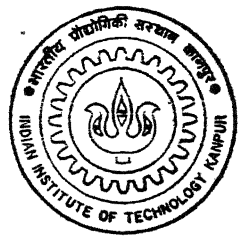
ARTIFICIAL NEURAL NETWORKS FOR FAULT DIAGNOSIS OF ROTATING MACHINERY

by

SATISH KUMAR

ME
1997
M
KUM
ART

TH
ME/1997/M
K96a



DEPARTMENT OF MECHANICAL ENGINEERING

Indian Institute of Technology Kanpur

DECEMBER, 1997

ARTIFICIAL NEURAL NETWORKS FOR FAULT DIAGNOSIS OF ROTATING MACHINERY

A thesis submitted
in partial fulfilment of the requirements
for the degree of
MASTER OF TECHNOLOGY

by

SATISH KUMAR . D.

DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR
DEC, 1997

POST GRADUATE
IIT Kanpur.
Submitted on 9/12/97

CERTIFICATE

It is certified that the work contained in this thesis entitled "**Artificial Neural Networks For Fault Diagnosis Of Rotating Machinery**", by Satish Kumar D has been carried out under my supervision and that this work has not been submitted elsewhere for a degree

Nalinaksh S. Vyas

23 Dec 1997

Nalinaksh S Vyas
Associate Professor
Department of Mechanical Engineering
Indian Institute of Technology
Kanpur 208016

ABSTRACT

A machine diagnosis scheme, generally makes use of a knowledge based technique to identify the reasons for departure of the machine from its normal behavior The present study aims to explore the possibility of employing Artificial Neural Networks (ANN) for condition monitoring applications in rotating machinery ANN offer a great deal of promise in such a monitoring strategy, especially in case of systems like Power Plants, where the number of components/systems are too many and complex to be mathematically modeled appropriately In the present study a user-interactive software for neural network simulator has been developed for prediction of faults in rotating machinery employing the back-propagation algorithm. Experiments were conducted on an existing laboratory rotoring Various types of faults were introduced in the set-up and processed signals were employed to train the neural network The adaptability of various neural network architectures has been investigated Neural Network training was carried out with the chosen architecture till desired degree of convergence is achieved The network is finally tested and validated for test data with unknown faults An overall success rate between 75-85 % is observed

ACKNOWLEDGMENTS

It gives me a great pleasure to place on record my deep sense of gratitude and sincere thanks to Dr Nalinaksh S Vyas my thesis supervisor, for his inspiring guidance and constant encouragement throughout the course of this work I am extremely thankful to him for suggesting a highly fascinating and impressive problem on *'Artificial neural networks for fault diagnosis of rotating machinery'* and instilling the confidence and independence in pursuing the work I am also thankful to Dr Raghuram for his valuable suggestions whenever needed

I am highly thankful to Ms Lalitha S who devoted her valuable time in many respects Thanks are due to Rajendra Magdum for his help in conducting the experiments, due to Mr A A Khan for his close association, due to Mr M M Singh for his service available whenever needed I am thankful to one and all who were in one way or other associated with me, towards the completion of my work

Indian Institute Of Technology, Kanpur
Dec, 1997

CONTENTS

NOMENCLATURE (i)

LIST OF FIGURES (ii)

LIST OF TABLES (iii)

1. INTRODUCTION 1

2. DIAGNOSIS OF ROTATING MACHINERY 3

2 1 Problems In Rotating Machinery 3
2 2 Analysis And Diagnosis Of Rotating Machinery 7

2 2 1 Time Domain Or Wave Form Analysis 7
2 2 2 Orbital Analysis 8
2 2 3 Spectrum Analysis 8
2 2 4 Cepstrum Analysis 9
2 2 5 Expert Systems For Fault Diagnosis 9
2 3 Remarks 10

3. FEATURES OF ARTIFICIAL NEURAL NETWORK 11

3 1 Model Of Neuron 12
3 2 Activation Function 14
3 3 Network Architecture 15
3 4 Learning Or Training Algorithms 16
3 5 The Backpropagation Algorithm 17

3 5 1 Forward Pass 18
3 5 2 Backward Pass 19
3 6 Remarks 24

4. NETWORK DESIGN FOR ROTORDYNAMIC APPLICATION 25

4 1 The Rotor Rig 25
4 2 Preparation Of Training Vector Set 27
4 3 Network Structure 36
4 4 Network Training With Moments Of $|z(t)|$ 38
4 5 Network Training With Moments Of $|dz(t)|$ 39
4 6 Network Training With Moments Of $|z(t)|\,|dz(t)|$ 40
4 7 Network Training With Cross Moments 42
4 8 Remarks 44

CONCLUSIONS 45

REFERENCES 46

NOMENCLATURE

x_i	Input signal to the network.
w_{kj}	Weight factor
u_k	Output from the summing junction
v_k	Modified output from the summing junction
$\varphi(\)$	Activation function
y_k	Output from the neuron
θ_k	Threshold
a	Logistic constant
$(x_{p1}, x_{p2},\ \ x_{pN})'$	Input vector to the network
net_{pj}^{h}	Net input values to the hidden layer unit
θ_L^h	Threshold for node L of the layer h
o_{pj}^{h}	Output from the hidden layer h for the node j
f_j^h	Activation function at node j of the hidden layer h
δ_{pk}	Individual error
E_p	Mean square error
η	(i) Learning Rate (ii) Mean
α	Moment
$\Delta_p w_{kj}^{o}$	Weight change at the output layer node j
$\Delta_p w_{ji}^{h}$	Weight change at the hidden layer node i
$p(x)$	Probability density function
$a_h(t)$	Acceleration in horizontal direction
$a_h(t)$	Acceleration in horizontal direction
$z(t)$	Complex time series
$dz(t)$	Differential complex time series
$E(\)$	Expectation operator
μ_k	Central moments
r_{xy}	2nd-order Cross moments
r_{xxy}	3nd-order Cross moments
l	Time lag

LIST OF FIGURES

Figure	Description	Page
3 1	Nonlinear Model Of Neuron	13
3 2	Modified Nonlinear Model Of Neuron	14
3 3	Sigmoid Function	15
3 4	Single Layer Feed Forward Network	16
3 5	Feed Forward Network With One Hidden Layer	16
3 6	Recurrent Network	16
3 7	One-Dimensional Lattice Of 3 Neurons	16
3 8	The Three Layer Backpropagation Network Architecture	18
3 9	Output Calculation At Each Node In The Forward Pass	19
3 10	Output Layer Weight Updating	21
3 11	Hidden Layer Weight Updating	23
4 1	Laboratory Arrangement Of Rotor Rig	26
4 2(a)	Vibration signal in time domain at 800 RPM for No-fault	28
4 2(b)	vibration signal in frequency domain at 800 RPM for No-fault	28
4 3(a)	Vibration signal in time domain at 800 RPM for play in coupling	28
4 3(b)	Vibration signal in frequency domain at 800 RPM for play in coupling	28
4 4(a)	Vibration signal in time domain at 800 RPM for Mass unbalance	29
4 4(b)	Vibration signal in frequency domain at 800 RPM for Mass unbalance	29
4 5(a)	Vibration signal in time domain at 800 RPM for bearing cap loose	29
4 5(b)	Vibration signal in frequency domain at 800 RPM for bearing cap loose	29
4 6(a)	Vibration signal in time domain at 800 RPM for Misalignment	30
4 6(b)	Vibration signal in frequency domain at 800 RPM for Misalignment	30
4 7(a)	Vibration signal in time domain at 800 RPM for Mass unbalance and Misalignment	30
4 7(b)	Vibration signal in frequency domain at 800 RPM for Mass unbalance and Misalignment	30
4 8(a-f)	Comparison Of First Five Moments Of *z(T)* For All Faults	31-32
4 9(a-f)	Comparison Of First Five Moments Of *dz(T)* For All Faults.	33-34
4 10(a-f)	Comparison Of Cross Moments For All Faults	34-35
4 11	The Error Graph With Moments Of *z(T)*	39
4 12	Comparision of error graphs for the network with 10 moments.	41
4 13	The Error Graph With The Cross Moments	43

LIST OF TABLES

Table	Description	Page
2 1	Identification And Correction Of Malfunctions Of Rotating Machinery	6
2 2	Diagnostic Techniques For Rotating Machinery	7
4 1	Training Data	37
4 2	Different Network Architectures And Testing Success For The Network With First Five Moment Of $z(t)$	39
4 3	Different Network Architectures And Testing Success For The Network With First Five Moment Of $dz(t)$	40
4 4	Performance Of (10,10,5) Architecture	40
4 5	Comparision Of Network Output With Actual Output For The Network With 10 Moments	42
4 6	Different Network Architectures And Testing Success For The Network With 14 Moments	43
4 7	Comparision Of Network Output With Actual Output For The Network With 14 Moments	44

1. INTRODUCTION

Supervision of machinery health is a subject of increased importance in the course of progressive automation Machinery condition monitoring is essential in order to anticipate problems in time so as to prevent complete failure It serves the important task of ensuring operation within design constraints Vibration analysis has proved to be a powerful tool for analysis of failure in rotating machinery Vibration analysis with appropriate instrumentation can provide reliable knowledge of machine condition and allows relatively inexpensive repair on a pre-scheduled controlled basis in comparison to catastrophic failure whose repair may cost much more Condition monitoring based on vibration signature analysis is a custom built system for effective product quality assurance or machinery fault diagnosis

Advances in data acquisition techniques and the developments in the areas of knowledge acquisition, processing and programming offer potential for the application of automated strategies for machinery process supervision which would permit early detection of faults than is possible through conventional limits and trend checks Meaningful and more efficient strategies for condition monitoring of rotating machinery are a pressing need, due to increasing demands on reliability and safety for power generating systems

There are three phases in the evolution of on-line computer-based vibration monitoring of rotating machinery The first phase includes collecting the vibration data and other operating parameters that are relevant Phase two consists of identifying the complicated but normal behavior of the system, accounting for the variation of operating parameters In phase three, use is made of knowledge based systems (expert systems) to identify the reasons for departures from normal behavior

Artificial Neural Networks (ANN) find their role in the third phase, of a monitoring strategy ANNs are also commonly referred to as Neural Network connections, Adaptive Networks, Neuron Computers and parallel distribution processors ANNs are massively

parallel interconnected networks of single processing elements intended to interact with the real-world in the same way as biological nervous system Neural Networks have been used to solve a wide variety of science and engineering problems that involve extracting useful information from complex or uncertain data In Power plant engineering, especially the fault diagnosis of various processes like nuclear power plant, chemical process control, monitoring of turbomachinery and design automation of engineering structures are the fields in which adoption of Neural Network performance is quiet promising

In the present study, a user-interactive software for a neural network simulator was developed for prediction of faults in rotating machinery The network employs back-propagation algorithm. It is written in ANSI-C and uses non-linear model of a neuron The simulator has an in-built input vector normalization Experiments were conducted on an existing laboratory rotor-rig Five different primary faults and their combinations were introduced in the experimental set-up The vibration signals collected through piezoelectric transducers from the bearing blocks were employed to train the network The adaptability of various neural network architectures was also investigated Neural network training was carried out with the chosen architecture till a desired degree of convergence is achieved The network was finally tested and validated for test data with unknown faults An overall success rate between 75-85 % was observed

A brief review of the commonly occurring faults in rotating machinery, currently available strategies for fault diagnosis and expert systems for fault diagnosis are discussed in Chapter 2 The basic principles of artificial neural networks principles and the back-propagation algorithm for fault diagnosis are described in Chapter 3 Chapter 4 describes the experimental work, signal processing, training and testing of the network and its validation

2. DIAGNOSIS OF ROTATING MACHINERY

The complexity of the dynamics of rotating machinery depends on the size as well as the operating speed On account of ever increasing demand for electric power and high speed transportation, the rotating machinery is being made increasingly light and flexible, thereby increasing the complexities of its dynamics Any fault in the system results in corresponding changes in the behaviour of the system and vibration signatures of the machine become important indicators of the health of the system Typical rotating machinery installations involve mechanical elements like rigid or flexible shafts carrying one or more discs with bladed components The arrangements may be supported in rolling element or fluid film bearings with associated components like coupling seals, gear-boxes, pulleys, etc The installation is carried by pedestals which are mounted on foundations The commonly occurring faults in rotating machinery are cracks on shafts, loss of blades on discs deteriorating balance, bows of shafting, shaft misalignment, oil film whirls, gear misalignment and damage, bearing eccentricities coupling inaccuracies and damage, pedestal and foundation cracks etc

2.1 Problems in Rotating Machinery

Rotordynamics has been an area of extensive research during the past two decades Reference can be made to the books by Rao(1992), Childs(1993), Dimentberg(1988) for a comprehensive discussion in the field, including fault classification and diagnosis Some of the problems in the rotating machines can be broadly listed as follows

Unbalance:

Unbalance in machinery arises due to eccentricities between geometrical and mass centers in components These unbalances generate large centrifugal forces in the machine which cause vibrations at the rotor speed e g one cycle per revolution Some specific unbalances created during the running of the machine give rise to additional characteristics

Distortion (casing and foundation):

Casing and foundation distortion cause vibration in an indirect way either by generating misalignment between driver and driven machines or by causing internal rub or uneven bearing contact This in turn transmits forces to the rotor, inducing it to generate forces of its own. such as unbalance and a wide variety of oil film and friction induced forces Another possibility is that the loads on casing supports shift and can set off a series of resonance problems Piping forces and foundation distortion often cause this type of difficulty

Misalignment:

Coupling misalignment causes friction or deflection forces in the coupling and this in turn causes the rotor and bearing system to deflect and to amplify these forces possibly creating secondary phenomenon such as harmonic resonance which can become severe Frequencies are characteristically $2\times$ running frequency (RF) for gear coupling, but $1\times$ RF is also often observed

Bending Resonance:

A rotor runs into resonance when the rotational speed is equal to its lateral (bending) vibration frequency Bending critical speeds can be easily detected by synchronous whirl conditions and fairly large amplitude at the rotor speed The critical speed region occurs over a fairly large range in the case of oil film bearings and is often accompanied by backward whirl Resonance of their structures, support and auxiliaries cause fairly large amplitudes of vibration at the rotor speed

Asymmetric shaft:

The response of the asymmetric shaft has several harmonics and the frequencies observed are $1\times$ rev, $2\times$ rev, $3\times$ rev and sometimes even harmonics, if the symmetry is predominant

Torsionals:

Torsional criticals in machinery are detected by torsional vibration measuring instruments Unless there is a gear box in the transmission unit, it is difficult to detect these criticals by

vibrometers or accelerometers used for conventional non-angular vibratory measurements Torsional criticals are predominant in reciprocating machinery, as there is a large excitation torque In rotating machinery, these are not easily excited, unless there is a disturbance arising out of gear transmission units Reciprocating machines have several criticals some of them major, between the starting and operating speeds

Oil-film whirl.

Instabilities in rotor can also arise due to oil film forces in the bearings This is self excited type of motion and is called oil whirl or oil whip The frequency of response at the onset of the whirl is always in the region of 0 4 to 0 5 × rev

Mechanical Looseness:

Looseness of components is the likely cause for most number of problems Loose rotor components such as disks, sleeves, thrust collars etc , cause internal friction problems Internal friction is a destabilizing factor The frequency of vibration is always the rotor critical speed and hence easy to detect Mechanical loose components like bolts, give rise to 1 × rev and harmonic frequency signals due to secondary phenomena The amplitude and phase continually change Loose assembly of bearings give rise to subharmonic response and the typical frequency response is 1/2 × rev and 1/3 × rev

Sub-harmonic Resonance:

This is a subsynchronous phenomenon that occurs at a frequency exactly equal to one-third one-fourth, or one-half of the operating speed One of the components of the nonlinear system usually vibrates at a natural frequency of the machine system, the resulting amplified vibration is termed a subharmonic resonance The condition can be corrected in several ways, by changing the natural frequency of the machine system, by removing the source of the nonlinear behavior (e g support pedestal, bearing), or by reducing the vibratory force that induces the nonlinear behavior

The most prominent type of rotating machinery problems, their symptoms and remedies are listed in Table 2 1 (Ehrich, 1992)

Table 2.1. Identification and correction of malfunction of rotating machinery

Fault	Frequency	Spectrum, Time-domain, Orbit shape	Correction
Mass unbalance	1X	Distinct 1X with much lower values of 2X, 3X etc , elliptical and circular orbits	Field or shop balancing
Misalignment	1X, 2X, etc	Distinct 1X with equal or higher values of 2X, 3X etc ,	Perform hot and/or cold alignment
Shaft bow	1X	Dropout of vibration around critical speed in Bode plot	Heating or peening to straighten rotor
Steam loading	1X	Load sensitive 1X	modify admission sequence repair diaphragms install nozzle blocks properly
Bearing wear	1X, subharmonics, orders	High 1X, high 0 5X, sometimes 1 5 or orders cannot be balanced	Replace bearing
Gravity excitation	2X	0 5 critical speed appears on Bode plot	Reduce eccentricity by balancing
Cracked rotor	1X, 2X	High 1X,0 5 critical speeds may show upon coast-down	Remove rotor
Looseness	1X plus large number of orders, 0 5X may show up	High 1X with lower-level orders large 0 5 order	Shim and tighten bolts to obtain rigidity
Coupling lockup	1X, 2X, 3X etc ,	1X with high 2X similar to misalignment start and stops may yield different vibration patterns	Replace coupling or remove sludge
Thermal instability	1X	1X has varying phase angle and amplitude	Compromise balance or remove problem
Oil-Whirl	0 35X to 0 47 X	Subsynchronous component less than 0 5 order informal loop in orbit	Temporary load bearing heavier, correct misalignment long term change bearing type
Oil-Whip	fn1	Subsynchronous component does not change with speed	Change bearing type
Subharmonic resonance	0 5X,1/3X, 0 25X, and higher	Subsynchronous Vibration depending on natural frequency	Remove looseness, excessive flexibility change natural frequency so it does not match fractional frequencies
Hysteresis	0 65X to 0 85X	High-magnitude fractional frequency(greater than 0 5x)	Eliminate or secure built-up parts
Rubs	0 25X,1/3X, 0 5X, or orders	External loops in orbit	Eliminate condition such as thermal bow and mass unbalance that cause rub
Trapped Fluid	0 8X to 0 9X	Bearing-sum and different frequencies	Remove fluid in rotor if possible, otherwise eliminate 1X component

1X=One time operating speed fn1=First natural frequency

2.2 Analysis and Diagnosis Of Rotating Machinery

Experimental and analytical techniques, both have been routinely used for diagnosing faults in operational rotating machinery These techniques are also used during the design and development processes Vibration analysis for fault diagnosis of large rotating machinery is finding increased application due to increased requirement of electric power and efficient transmission of power between shafts in power plants Parameter identification techniques also find wide use to quantify natural frequencies, mode shapes, damping stability and frequency-response characteristics Some of these diagnostic techniques are listed in Table 2 2

Table 2.2. Diagnostic Techniques for Rotating Machinery

Technique	Use	Description	Instrument Type
Time-domain analysis	Modulation, Pulses, Phase, Truncation glitch	Amplitude versus time	Analog and digital oscilloscope, FFT spectrum analyzer
Orbital analysis	Shaft motion subsynchronous whirl	Relative displacement of rotor bearing in X Y directions	Digital vector filter, Oscilloscope
Spectrum analysis	Direct frequencies, natural frequencies, sidebands, beats, subharmonics, sum and difference frequencies	Amplitude versus frequency	FFT spectrum analyzer
Cepstrum analysis	Sideband and harmonic frequency measurement, accurate quantification of sideband and harmonic severity	Inverse Fourier transform of logarithmic power spectrum	FFT spectrum analyzer

2.2.1 Time Domain Or Wave Form Analysis:

Time domain signal is an unprocessed record of the vibration as picked up from a transducer It consists of information about the physical behavior of machine A serious limitation is that it can be too complex for analysis if excessive noise, signal modification or several frequencies are present The processing can be done, but it should be minimized whenever possible so that errors are not introduced into the data In the time domain, frequency and phasing are two important factors that are to be evaluated for fault diagnosis Valuable information is provided by signal shape, i e truncation, pulses, modulation, glitch or shaft-induced signals obtained from a proximity probe that are caused by scratches on the shaft Time between events represents the frequency

components within the machine The phase between two signals provides information about vibratory behavior that can be used to diagnose a fault such as misalignment

2.2.2 Orbital Analysis:

The horizontal and vertical motions of the rotor with respect to a sensor mounted on the bearing are simultaneously displayed on oscilloscope This is called the orbit plot The orbit represents the instantaneous position of the rotor The orbit can be used to access oil whirl and other asynchronous motions as well as synchronous phenomena such as mass unbalance and misalignment Orbital analysis is possibly the best way to distinguish between mass unbalance and misalignment, both of which cause once-per revolution force

2.2.3 Spectrum Analysis:

Spectrum analysis is conducted with an FFT algorithm or filters The processed signal gives information about the amplitude and phase content at various frequencies Frequencies of vibration response can be related to direct excitation frequencies or their orders, natural frequencies, sidebands, subharmonics and sum-difference frequencies Frequencies can be identified directly in a linear system, because the frequencies of the measured vibration response are equal to those of the forces causing the vibration When a minimum number of frequencies are present, signal-resolution is required, and electric filters can be used to analyze the spectrum. The operating speed of the machine is usually the fundamental frequency used in the analysis The frequency at operating speed is identified and then linked to other frequencies in the spectrum such as direct orders (multiple) of operating speed These multiple orders are due to a variety of faults that are present in the rotating machinery The exact fault can be predicted by comparing with good test data that is directly proportional to the information available on the design of a machine and its working mechanisms This is especially true when similar frequencies are obtained for different faults, e g mass unbalance and misalignment

2.2.4 Cepstrum analysis:

Cepstrum is defined as the inverse Fourier transform of the logarithmic power spectrum commonly used in vibration analysis The complex cepstrum is reversible to a time signal The terminology used in cepstrum analysis is analogous to that used in spectrum. A cepstrum domain consists of data processed one step further than the frequency domain A cepstrum is effective for accurately measuring frequency spacing, harmonic and sideband patterns in the power spectrum. One component of the cepstrum represents the global power content of an entire family of harmonics or sidebands The severity of a defect such as that in a rolling element bearing is provided by one component of the cepstrum display

2.2.5 Expert Systems For Fault Diagnosis

Conventional manual trend checking methods and the above discussed diagnosing procedures have their own stipulations and often require human experts to analyze the vibration signature, which are expensive and do not yield the exact prediction of faults This has motivated the development of Expert systems or Knowledge based systems for accurate fault diagnosis of rotor dynamic installations, which are cost effective and time saving Expert systems are intelligent computer programs that are based on well defined algorithms which exhibit within a specific domain, a degree of expertise as a prediction system, diagnostic system, design system, planning system, speech recognition system, monitoring system etc These are often considered to be one of the most popular application of artificial intelligence (AI) Artificial neural networks (ANN) are often thought of as distinct from the AI, though the ANNs are common in AI literature ANNs simulate the biological process of human brain and nervous system, whereas AI programs attempt to identify and execute higher level process and relationship that are often based on logic and linguistics

Knowledge base is the most important part of any expert system. It is stored in the form of facts and rules The facts are termed as deep knowledge and rules are simply the heuristics The knowledge that is stored is controlled by an inference engine The inference engine interacts with the user and uses the facts in the knowledge base according to the

rules contained in it Since the knowledge or data in most cases may be incomplete or uncertain the models employ probabilistic reasoning technique such as Bayes's rule fuzzy logic Dempster-Shafer calculus etc Reference can be made to the work of Nema (1994), who employed probabilistic reasoning techniques for the development of an Expert Shell for diagnosis of rotating machinery ANNs are increasingly becoming popular due to their capabilities of organizing and identifying uncertain patterns This feature of the ANN, motivates the researchers to diagnose the various malfunctions of the machinery

2.3 Remarks

In the present study an ANN is developed to diagnose the rotating machinery malfunctions The knowledge is acquired through standard learning techniques Each segment of the acquired knowledge gets a weightage which is related to its importance These weights derive a functional relation between input and output patterns, during the learning process

The relevant features of Artificial Neural Networks are described in the next chapter

3. FEATURES OF ARTIFICIAL NEURAL NETWORKS

An Artificial Neural Network can be defined as a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use It resembles the brain in two respects (i) knowledge is acquired by the network through a learning process and (ii) inter-neuron connection strengths, known as synaptic weights are used to store the knowledge

Artificial Neural Networks technology is of a relatively recent origin and very few studies are available which relate to vibrations and dynamics of machinery Most of the work has been carried out in the present decade Mayes (1994) applied ANN for on-line vibration monitoring of large turbo-generators The investigations focused on data processing and the use of the neural networks were discussed Elkordy, Chang and Lee (1994) investigated the applicability of ANN for vibration signature analysis of a five-storey steel structure The primary investigations showed that ANNs have considerable potential to assess structural damage Trzynadlowski, Karam, Gandikota and Ghassemzadeh (1994) applied the neural net software for the validation and recovery of the distorted vibration of electromachine systems Vibration signals, which represent the lateral displacement of a shaft, were obtained from proximity probes and the Discrete Fourier coefficients were calculated These were used as the inputs to the network to classify fault and no-fault signal Fuzzy-input neural net adaptive expert systems were presented by Dimargonas (1995) for rotor diagnosis and prognosis A traditional binary classification system was adopted to diagnose the fault in rotating machinery The two possible states, fault and no-fault were described by 1 or 0 Available knowledge for failure diagnosis in turbomachinery was utilized to initially teach the system Ahn and Cho (1996) proposed a new vibration control scheme using the ANN along with electromagnetic and pneumatic principles as a hybrid type active vibration isolation system. The characteristics were investigated via computer simulation as well as experimentation The proposed control scheme could suppress the transmissibility of the vibration isolation system to below 0 63 over the entire frequency range, including the resonance frequency, without complex

calculation or prior manipulation Haung and Lian (1996) proposed a different hybrid scheme with a combination of fuzzy logic and neural network algorithms for active vibration control A fuzzy logic controller was designed for controlling the main influence part of the MIMO (multi-input/multi-output) system Very recently, Han Xiu Wang Chen and Tan (1997) presented the adaptation of neural network for fast-valve controlling of a power generation plant The backpropagation neural network was used to train the feed-forward neural network controller The computer simulated backpropagation network results were compared with the conventional fast valving methods applied to the same system The investigations proved that the ANN controller, has satisfactory generalization capability reliability and accuracy for the critical control operation McCormick and Nandi (1997) contemplated the application of ANN for real-time classification of rotating shaft conditions The vibration signals were collected from the machinery in time-domain Simple signal processing techniques were applied to prepare the training vector set The use of the ANN was described to classify the load and no-load of the rotating machinery The network results were compared with the frequency domain analysis

The salient features of Artificial Neural Networks are described in this chapter

3.1 Model Of A Neuron

A Neuron is an information processing unit that is fundamental to the operation of a neural network Fig 3 1 shows the model of a Neuron The three basic elements of the neuron model are described below (Haykin, 1994)

A Set of Synapses

Synapses are connecting links, each of which is characterized by a weight or strength of its own It specifies the connection between a signal x_j at the input of the sample j, and a Neuron k The weighting factor is w_{kj}

An Adder

Adder sums up the input signals weighted by the respective synapses of the Neuron The operation is similar to that of a linear combiner

An Activation Function

Activation Function limits the amplitude of the output of the Neuron The activation function is also referred to as a *squashing* function

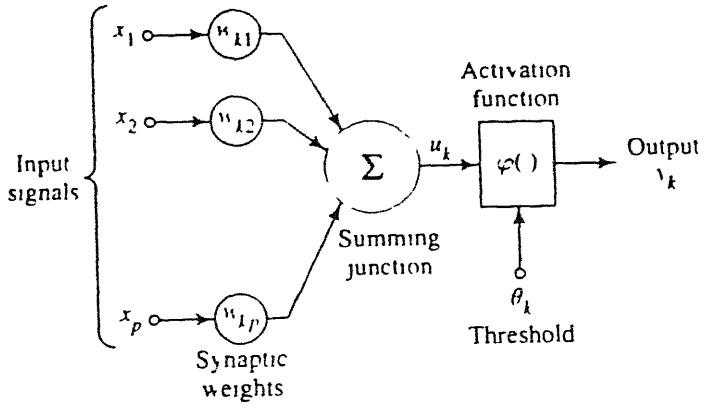


Fig 3 1 Nonlinear Model Of Neuron

As shown in the above figure (Fig 3 1), the model of a Neuron also includes an externally applied threshold function θ_k, that has the effect of lowering the net input of the activation function On the other hand, the net input of the activation function may be increased by employing a bias term rather than a threshold The bias is the negative of the threshold In mathematical terms a neuron k is described by

$$u_k = \sum_{j=1}^{p} w_{kj} x_j \ , \qquad y_k = \varphi\ (u_k - \theta_k) = \varphi(v_k) \tag{3 1}$$

where $x_1, x_2 \quad x_p$ are the input signals, $w_{k1}, w_{k2} \quad w_{kp}$ are the synaptic weights of the neuron k u_k is the linear combiner output, θ_k is the threshold, $\varphi(\)$ is the activation function and y_k is the output signal of the neuron v_k is the output from the neuron after summation

Equations (3 1) can be equivalently written as

$$v_k = \sum_{j=0}^{p} w_{kj} x_j \tag{3 2}$$

$$y_k = \varphi(v_k) \tag{3 3}$$

In equation (3 2) a new synapse has been added, whose input and weight, respectively are

$$x_0 = -1 \qquad w_{k0} = \theta_k \tag{3 4}$$

and therefore the model of the Neuron k can be reformulated as in Fig 3 2

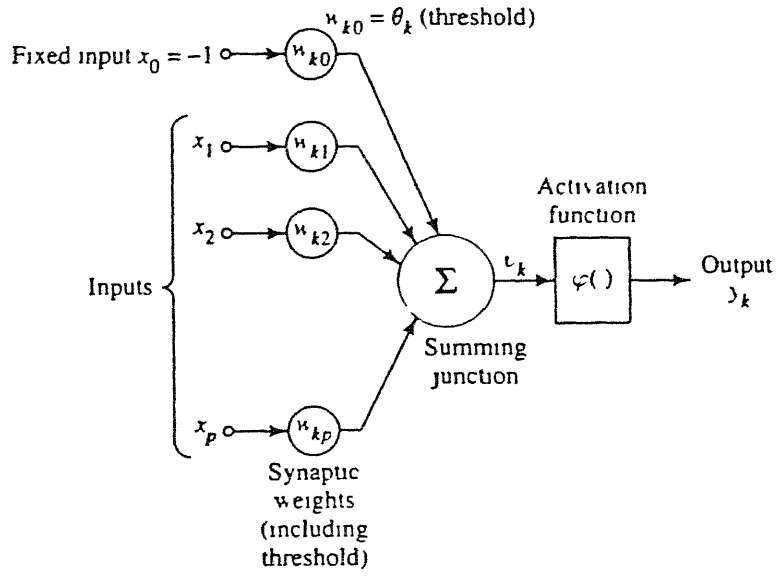


Fig 3 2 Modified Nonlinear Model Of Neuron

3.2 Activation Function

The activative function denoted by $\varphi(\)$ defines the output of a Neuron in terms of the activity level at its input The sigmoidal function is by far the most common form of activation function used in the construction of the artificial neural networks It is defined as a strictly increasing function that exhibits smoothness and asymptotic properties An example of the sigmoid is the logistic function defined by (Fig 3 3)

$$\varphi(v) = \frac{1}{1+\exp(-av)} \tag{3 5}$$

where a is the slope parameter of the sigmoid function

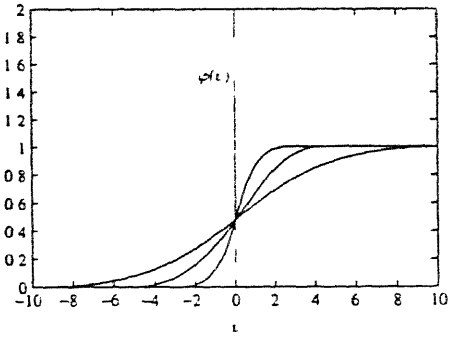


Fig 3 3 Sigmoid Function

3.3 Network Architectures

The manner in which the neurons in a neural network are structured is intimately linked with the learning algorithm used to train the network Four general classes of network architectures are

i) Single layer feed forward network

This is a network of neurons organized in the form of layers The simplest form of a layered network consists of an input layer of source nodes. that projects on to an output layer of neurons but not vice-versa This network is of feed-forward type (Fig 3 4)

ii) Multi-layer feed forward network

The second class of feed-forward network distinguishes itself by the presence of one or more hidden layer The function of hidden neurons is to intervene between the external input and the network output (Fig 3 5)

iii) Recurrent networks

A recurrent network distinguishes itself from the feed-forward neural network in that it has at least one feedback loop For example, a recurrent network may consist of single layer of neurons, with each neuron feeding its output signal back to the input of all other neurons (Fig 3 6)

vi) Lattice structure

A lattice consists of a one-dimensional, two-dimensional or higher dimensional array of neurons with a corresponding set of source nodes that supply the input signals to the arrays (Fig 3 7) The dimension of the lattice refers to the number of dimensions in which the graph lies

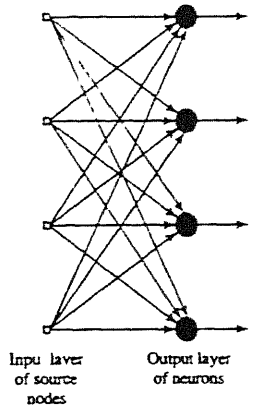


Fig 3 4 Single layer Feed forward network

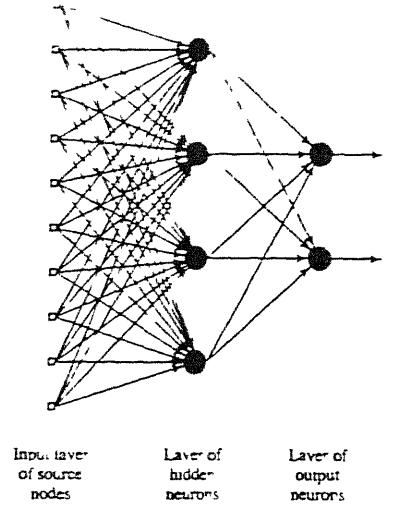


Fig 3 5 Feed forward network with one hidden layer

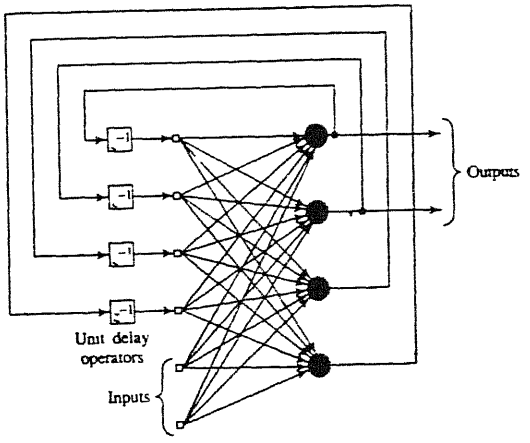


Fig 3 6 Recurrent network

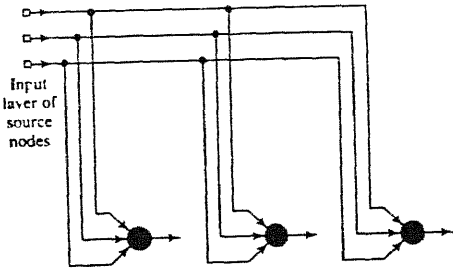


Fig 3 7 One-dimensional lattice of 3 neurons

3.4 Learning Or Training Algorithms

Most training techniques, for ANNs have been developed from common roots and share many characteristics A brief description of some of them is given below

Supervised And Unsupervised Learning

Training algorithms are classified as supervised or unsupervised In supervised training, there is a teacher that presents input pattern to the network, compares the resulting output with those desired, and then adjusts the network weights in such a way as to reduce the difference However, in unsupervised training there is no necessity for the teacher Input patterns are applied, and the network self-organizes by adjusting its weights according to

well-defined algorithms Most of the today's training algorithms have evolved from the concepts of Hebb (1961), which involve adjusting the network weights according to the product of the excitation level of the source and destination neurons Mathematically.

$$w_{ij}(t+1) = w_{ij}(t) + \eta out_i out_j \qquad (3\ 6)$$

where, $w_{ij}(t)$ and $w_{ij}(t+1)$ are the weights from neuron i to neuron j, prior and after adjustment, respectively η is the learning-rate coefficient , out_i is the output of neuron i and input to neuron j, while out_j is the output of neuron j

3.5 Backpropagation Algorithm (BPA)

BPA is found to be the most common learning algorithm. It has been tested with a number of different problems and has been found to perform well in most cases It is designed to solve the problems of choosing weight values for layered artificial neural networks with feed-forward connections from input layer to the hidden layer and then to the output layer The BPA performs the input to output mapping by minimizing a cost function using a gradient search technique The cost function (which is equal to the mean squared difference between the desired and the actual net output) is minimized by making wide connection adjustments according to the error between the computed and target output processing element values There are two stages in the development of a backpropagation algorithm - namely forward pass and backward pass During the forward pass all the weights of the network are initialized randomly and the network outputs and the difference between the actual and target output (i.e error) is calculated for the initialized weights During the backward step, the initialized weights are adjusted to minimize the error by propagating the error backwards The network outputs and error are calculated again with the updated weights and the process repeats till the error is acceptably small The two steps are described below (Fig 3 8)

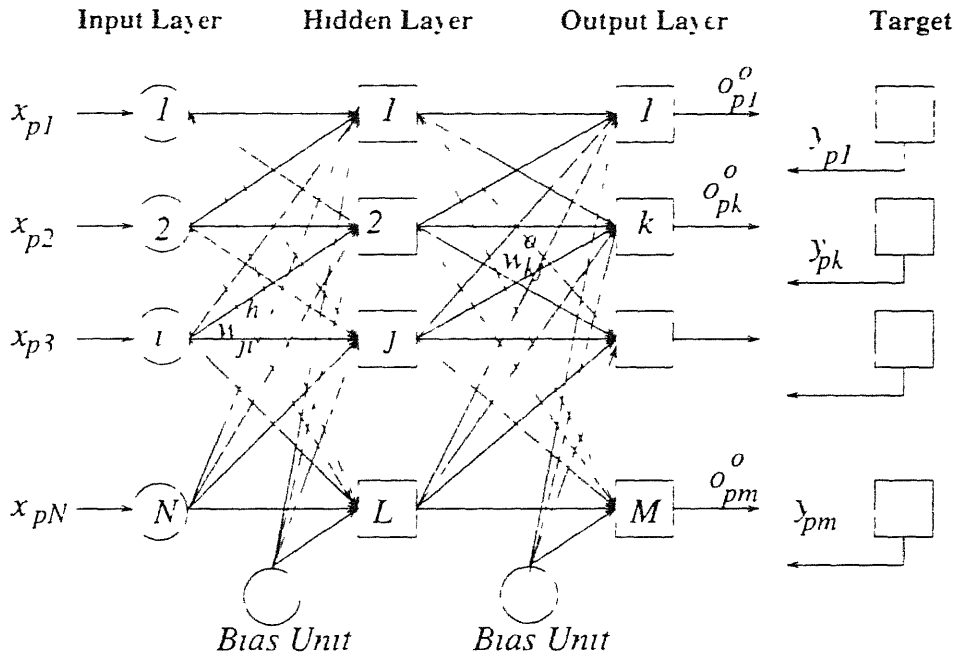


Fig 3 8 The three layer Back propagation network Architecture

3.5.1 Forward Pass

The input vector p to the network is $x_p = (x_{p1} \quad x_{pN})'$ (3 7)

where x_{pi} represents the input attribute i for the vector p

The net input values to the hidden layer units are

$$net_{pj}^h = \sum_{i=1}^{N} w_{ji}^h x_{pi} + \theta_j^h = \begin{bmatrix} w_{11}w_{12} & w_{1n} \\ w_{21}w_{22} & w_{2n} \\ & \\ w_{L1}w_{L2} & w_{Ln} \end{bmatrix} \begin{bmatrix} x_{p1} \\ x_{p2} \\ \\ x_{pr} \end{bmatrix} + \begin{bmatrix} \theta_L^1 \\ \theta_L^2 \\ \\ \theta_L^h \end{bmatrix} \quad (3\ 8)$$

w_{ji}^h represents the weight of the layer h from node i node to node j and θ_L^h represents the threshold for the node L of the layer h

The outputs from the hidden layer(which is input to the output layer) are

$$o_{pj}^h = i_{pj} = f_j^h(net_{pj}^h) \quad (3\ 9)$$

o_{pj}^h is the output from the node j of the hidden layer h, f_j^h is the activation function at node j of the hidden layer h

The equations (3 8) (3 9) are represented graphically in Fig 3 9

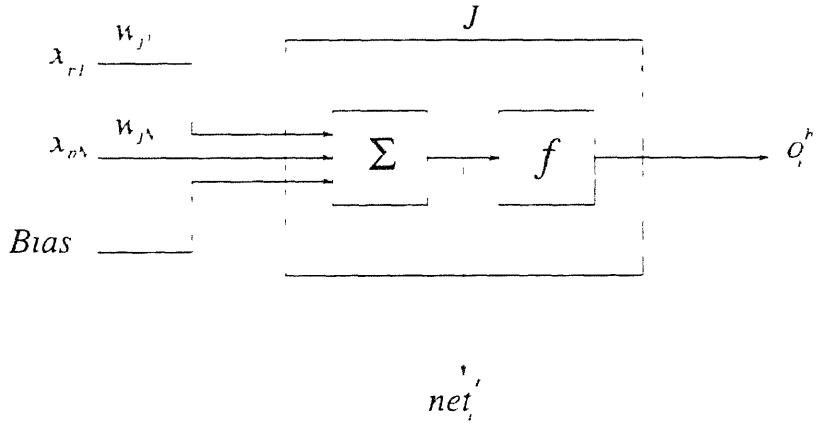


Fig 3 9 Output calculation at each node in the forward pass

In the above the net-input values at output layer unit are

$$net^{o}_{pk} = \sum_{j=1}^{L} w^{o}_{kj} i_{pj} + \theta^{o}_{k} \qquad (3\ 10)$$

the outputs at output layer units are $o^{o}_{pk} = f^{o}_{k}(net^{o}_{pk})$ (3 11)

the individual error at each output unit is $\delta_{pk} = y_{pk} - o^{o}_{pk}$ (3 12)

and the mean squared error at all output unit is $E_p = \frac{1}{2}\sum_{k=1}^{M} \delta_{pk}^{\ 2}$ (3 13)

3.5.2 Backward Pass

The weights at the output and hidden layer are adjusted to minimize the mean square error in the backward pass While the output layer nodes are associated with the target output and the nodes at the hidden layer are not, the weights updating at the output and hidden nodes is different These two procedures are described below

Updatmg The Output Layer Weights

From equation (3 12) and (3 13) E_p can be expressed as

$$E_p = \frac{1}{2}\sum\left(y_{pk} - o^o_{pk}\right)^2 = \frac{1}{2}\sum\left\{y^o_{pk} - f^o_k\left(net^o_{pk}\right)\right\}^2 \tag{3 14}$$

The weight change of an output layer weight is the negative gradient of E_p with respect to output layer weights w^o_{kj}

$$\frac{\partial E_p}{\partial w^o_{kj}} = -(y_{pk} - o^o_{pk})\frac{\partial f^o_k}{\partial(net^o_{pk})}\frac{\partial(net^o_{pk})}{\partial w^o_{kj}} \tag{3 15}$$

However, from equation (3 10)

$$\frac{\partial(net^o_{pk})}{\partial w^o_{kj}} = \left(\frac{\partial}{\partial w^o_{kj}}\sum w^o_{kj} i_{pj} + \theta^o_k\right) = i_{pj} \tag{3 16}$$

and from equation (3 11)

$$\frac{\partial f^o_k}{\partial\left(net^o_{pk}\right)} = f^o_k\left(net^o_{pk}\right) = o^o_{pk}\left(1 - o^o_{pk}\right) \tag{3 17}$$

Therefore, the weight change at the output layer weight is

$$\Delta^o_p w = \frac{-\partial E_p}{\partial w^o_{kj}} = (y_{pk} - o^o_{pk}) f^o_k\,(net^o_{pk}) i_{pj} \tag{3 18}$$

Now denotmg

$$\delta^o_{pk} = (y_{pk} - o^o_{pk}) f^o_k\,(net^o_{pk}) \tag{3 19}$$

the weight change at the output layer weights can be written as

$$\Delta^o_p w = \delta^o_{pk} i_{pj} \tag{3 20}$$

In practice, mstead of directly applying the above weight change to the weight at the output layer, two network parameters are mtroduced, namely, Learning Rate Coefficient η and Momentum α, m order to make the learning progress smooth and to ensure that the weight changes take place always m the same direction

$$w^o_{kj}(t+1) = w^o_{kj}(t) + \eta(y_{pk} - o^o_{pk}) f^{o'}_k\,(net^o_{pk}) i_{pj} + \alpha w^o_{kj}(t-1) \tag{3 21}$$

The selection of the learnmg rate parameter η, has a significant effect on the network performance Usually, η is a small number, of the order of 0 05 to 0 9 A small value of η implies that the network will have to make a large number of iterations It is often possible

to increase the value of η as network error decreases thereby increasing the speed of convergence towards the target output The other way to increase the convergence speed is by adopting an extra momentum term α (equal to a fraction of the previous weight change $\Delta_p w$) while updating the weights This additional term tends to keep the weight changes in the same direction The entire weight updating process at output layer can be represented graphically by Fig 3 10

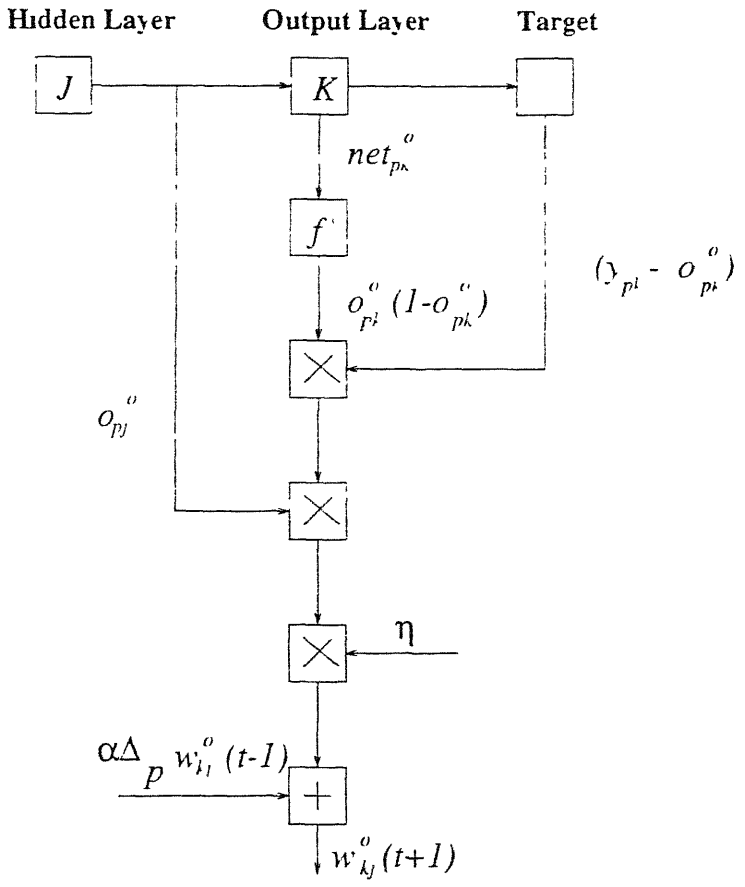


Fig 3 10 Output layer weight updating

Updating The Hidden Layer Weights

The adjustment of weights for the hidden layer nodes is proportional to their individual initial contributions there is no target output as in the case of the output layer From equations (3 10) and (3 11)

$$E_p = \frac{1}{2}\sum_k \left(y_{pk} - f_k^o \left\{ \sum w_{kj}^o I_{pj} + \theta_k^o \right\} \right)^2 \tag{3 22}$$

The weight change of a hidden layer weights is the negative gradient of E_p with respect to hidden layer weights w_{ji}^h

$$\frac{\partial E_p}{\partial w_{ji}^h} = -\sum_k \left(y_{pk} - o_{pk}^o \right) \frac{\partial o_{pk}^o}{\partial (net_{pk}^o)} \frac{\partial (net_{pk}^o)}{\partial I_{pj}} \frac{\partial I_{pj}}{\partial (net_{pj}^h)} \frac{\partial (net_{pj}^h)}{\partial w_{ji}^h} \tag{3 23}$$

From the equations (3 10) to (3 14) the individual terms can be expanded as

$$\frac{\partial o_{pk}^o}{\partial (net_{pk}^o)} = \frac{\partial \left(f_k^o \left(net_{pk}^o \right) \right)}{\partial (net_{pk}^o)} = f_k^o \left(net_{pk}^o \right) = o_{pk}^o \left(1 - o_{pk}^o \right) \tag{3 24}$$

$$\frac{\partial (net_{pk}^o)}{\partial I_{pj}} = \frac{\partial \left(\sum_{j=1}^{L} w_{kj}^o I_{pj} + \theta_k^o \right)}{\partial I_{pj}} = w_{kj}^o \tag{3 25}$$

$$\frac{\partial I_{pj}}{\partial (net_{pj}^h)} = \frac{\partial \left(f_j^h net_{pj}^h \right)}{\partial (net_{pj}^h)} = f_j^h \left(net_{pj}^h \right) = o_{pj}^h \left(1 - o_{pj}^h \right) \tag{3 26}$$

$$\frac{\partial (net_{pj}^h)}{\partial w_{ji}^h} = \frac{\partial \left(\sum_{i=1}^{N} w_{ji}^h x_{pi} + \theta_j^h \right)}{\partial w_{ji}^h} = x_{pi} \tag{3 27}$$

Substituting the above equations (3 24 to 3 27) in equation (3 23)

$$\Delta_p w_{ji}^h = -\frac{\partial E_p}{\partial w_{ji}^h} = x_{pi} o_{pj}^h \left(1 - o_{pj}^h \right) \sum_k \left(y_{pk} - o_{pk}^o \right) o_{pk}^o \left(1 - o_{pk}^o \right) w_{kj}^o \tag{3 28}$$

Network parameters η, α can be introduced in a manner similar to that in the case of the output layer, to express the final weight change at the hidden layer as

$$w_{ji}^h (t+1) = w_{ji}^h (t) + \eta x_{pi} o_{pj}^h \left(1 - o_{pj}^h \right) \sum \left(y_{pk} - o_{pk}^o \right) o_{pk}^o \left(1 - o_{pk}^o \right) w_{kj}^o + \alpha w_{ji}^h (t-1) \tag{3 29}$$

The entire weight updating process at hidden layer is represented graphically in Fig3 11

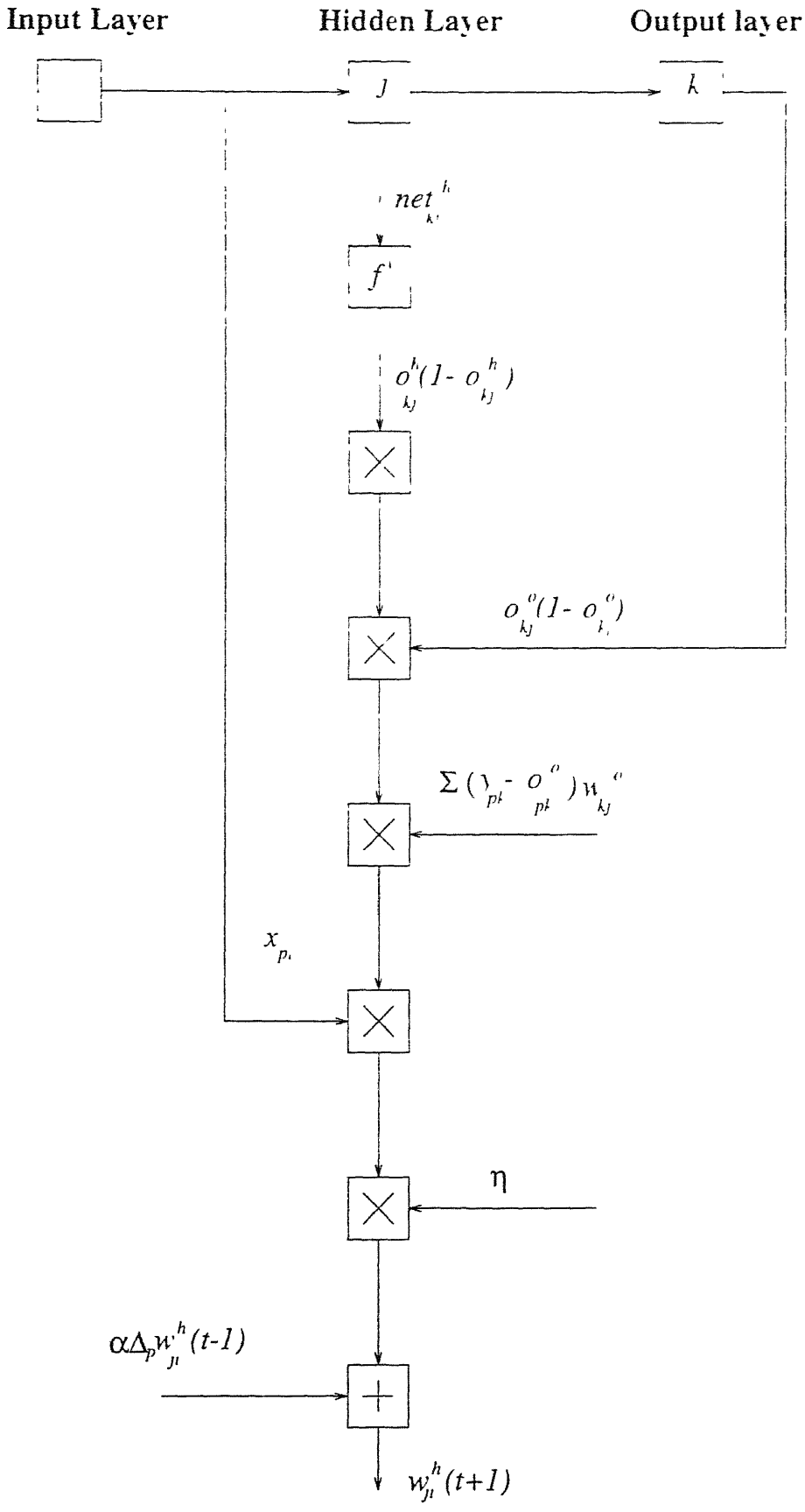


Fig 3 11 Hidden layer weight updating

3.6 Remarks

The ANN algorithm developed in this study employs the backpropagation learning technique The training is done with processed data until adequate convergence of the network was achieved and the stored weights are used to estimate the output patterns corresponding to the given input patterns, during pattern recognition The algorithm is written in ANSI-C

4. NETWORK DESIGN FOR ROTORDYNAMIC APPLICATIONS

An effective machine diagnosis requires that the diagnosis algorithm is trained for all the fault conditions that need to be identified Availability of such training data, and generation of such data in cases where it is not readily available is the primary requirement of any diagnosis scheme Separate data sets which are not part of the training sets are required for validation and testing of the network The training and the test data, during the present study were generated on a laboratory rotor-rig The rig, data generation, training and testing of the algorithm are described in this chapter

4.1 The Rotor Rig

The Rig (Fig 4 1) consists of a 10 mm diameter shaft carrying a centrally located steel disc weighing 0 5 kg It is supported in identical rolling element bearings (type 6200 SKF ball bearings) at two ends and driven by an 50 Watts, 230 Volts A C /D C electric motor through a flexible spider coupling The following faults were deliberately introduced in the rig for generating training data

- **Rotor with no-fault**

 The rotor was balanced, the alignments and fittings were done properly so as to *presumably* classify it as a system with no-fault

- **Rotor with mass unbalance**

 A mass of about 0 05 Kg was added at radius of 25 mm on the rotor disc and unbalance was created

- **Rotor with bearing cap loose**

 The cap on the bearing block was loosened so as to create a gap of approximately 1 mm. between the outer race of the bearing and the cap of the bearing block No attempt was made to quantify the fault accurately

CENTRAL LIBRARY
I I T KANPUR
Acc No A 124919

- **Rotor with misalignment**

 Misalignment in the rotor was created by shifting the bearing block sideways by about 3 mm, so that the axis of the two bearing blocks were out of alignment by about 3 mm. Again no attempt was made to quantify this fault in strict numerical terms.

- **Play in spider coupling**

 The flexible rubber spider in the coupling was removed and a small cut was introduced such that a radial clearance was created at the two halves of the coupling at the outer diameter of 15 mm. This fault was also not quantified in numerical terms.

- **Rotor with both mass imbalance and misalignment**

 In this case mass unbalance and misalignment were both introduced simultaneously in the rig.

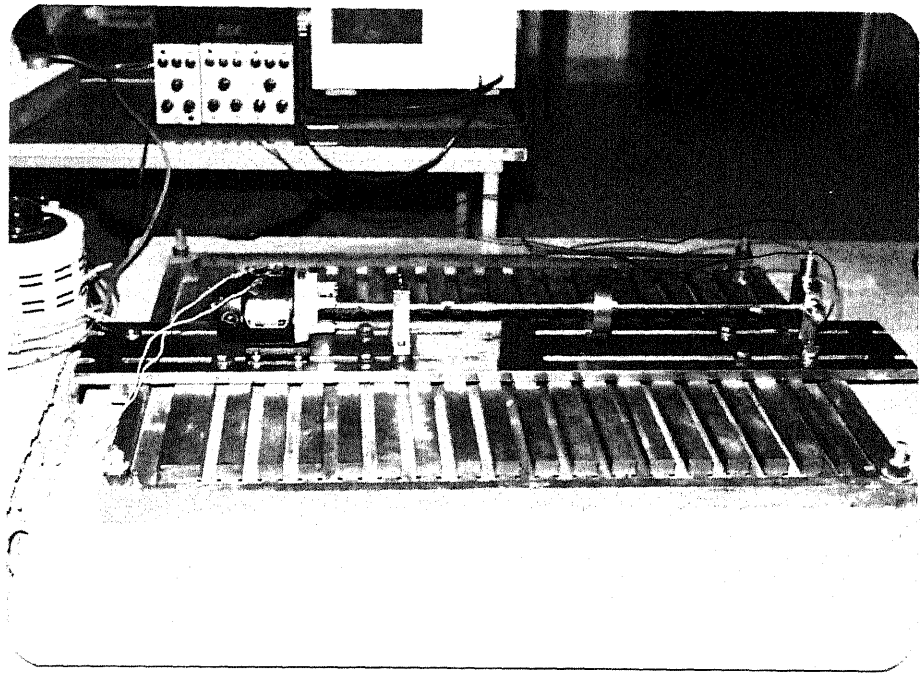


Fig 4.1. Laboratory arrangement of rotor rig

These faults were introduced one at a time and the vibration signals of the rotor wer picked up from the bearing-block caps by piezo-electric accelerometers (Burel and Kjae make, charge sensitivity = 10.20 $pc/m/s^2$), for a range of rotor speeds from 500 to 90 rpm. The signals were amplified by charge amplifiers (Make 2635, Bruel and Kjaer). Th

amplified signals were stored in floppy diskettes using a portable dual channel FFT analyzer(Ono-Sokki) for further processing Typical rotor vibrations sensed at the bearing cap, by the accelerometer for each of the six faults introduced in the rotor, are shown in Figures (4 2) to (4 7) The signals pertain to a typical rotor speed of 800 RPM The FFT of the time signal is also shown in these figures Twenty such vibration signals were taken for each of the faults introduced at every speed of rotation Signals were obtained for five rotor speeds in the range 500-900 RPM These signals were processed further, to form inputs to the training algorithm.

4.2 Preparation Of The Training Vector Set

It is evident from the FFT of Figs 4 2(b) to 4 7(b), that there can be a large number of inputs (say if amplitude and phase information is used as training data, they need to be provided for a reasonably large number of frequencies, to retain all the relevant features of the signal) Other high resolution methods, such as autoregressive analysis, time frequency distribution methods e g wavelet transformation, also generate a large number of inputs In order to reduce the number of inputs to the network it is necessary to extract stationary features of the vibration signals An efficient way to reduce the number of inputs, while retaining most of the relevant information, is to extract time-invariant features from the stationary time series of Figs 4 2(a) to 4 7(a) This is easily done by estimating the zero-lag moments (McCormick and Nandi 1997) Since moments of time series characterize the probability density function $p(x)$ of the vibration signal for each condition, the following different moments are calculated for the complex time series given by

$$z(t) = a_h(t) + ja_v(t) \tag{4 1}$$

and differential complex time series given by

$$dz(t) = \frac{dz}{dt} \approx z(t) - z(t-1) \tag{4 2}$$

where $a_h(t)$ is the vibration signal from the bearing in the horizontal direction and $a_v(t$ is the signal in the vertical direction

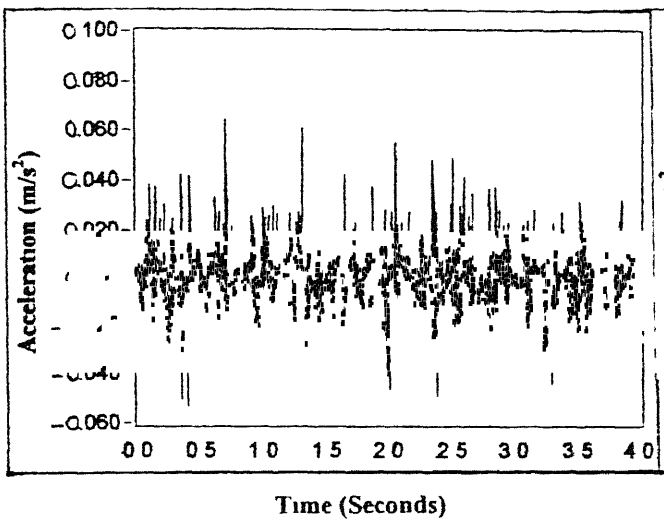


Fig 4 2 (a) Vibration signal in time domain at 800 RPM for No-fault

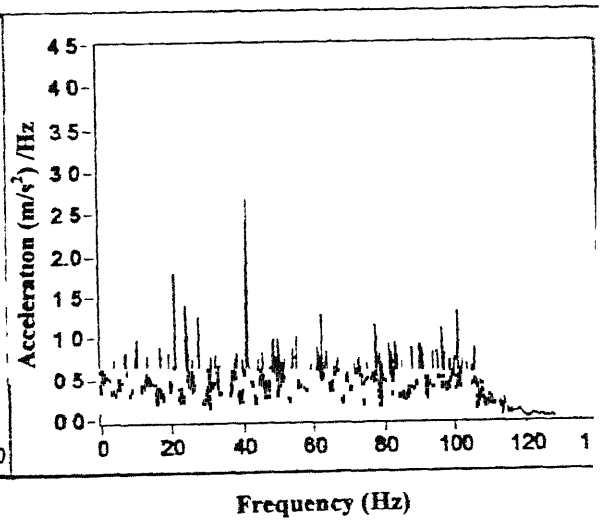


Fig 4 2 (b) Vibration signal in Frequency dom at 800 RPM for No-fault No-fault

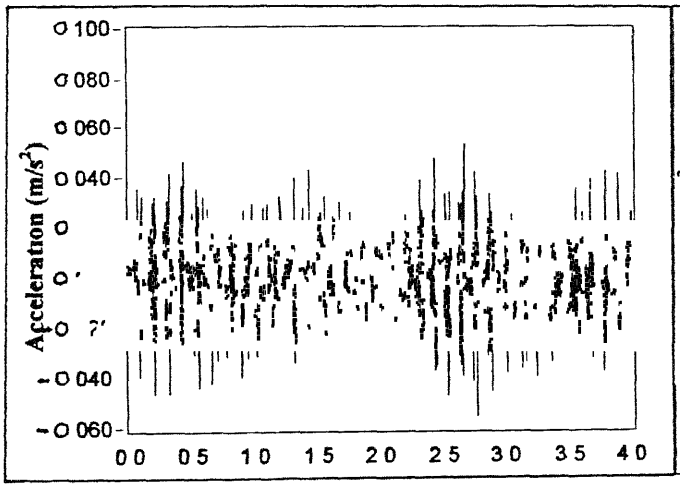


Fig 4.3 (a) Vibration signal in time domain at 800 RPM for Play-in-coupling

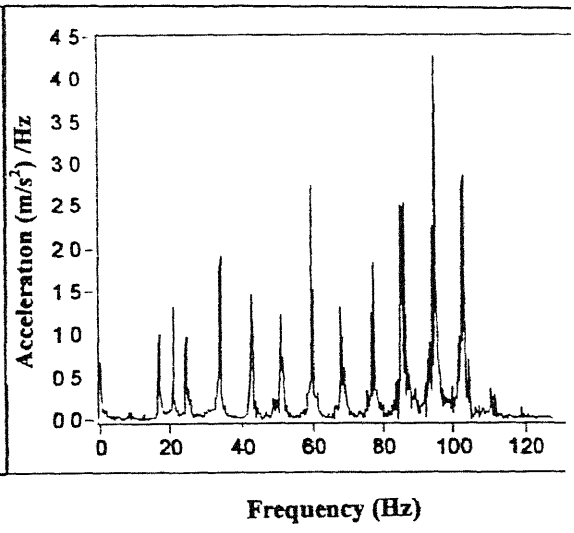


Fig 4.3 (b) Vibration signal in Frequency domain at 800 RPM for Play-in-coupling

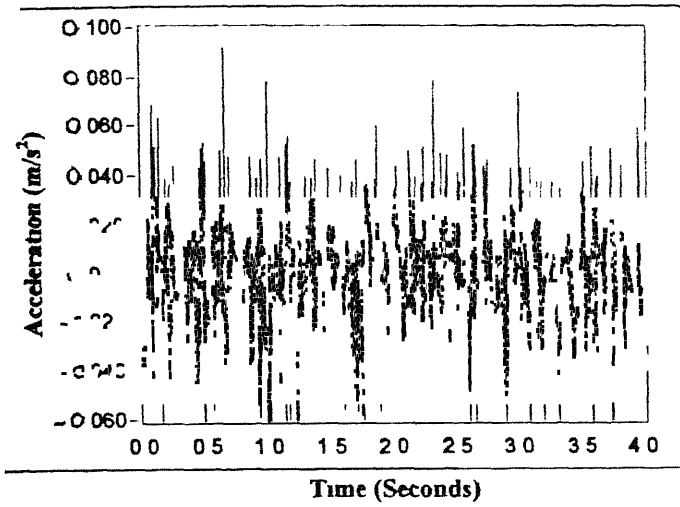


Fig 4 4 (a) Vibration signal in time domain at 800 RPM for Mass unbalance

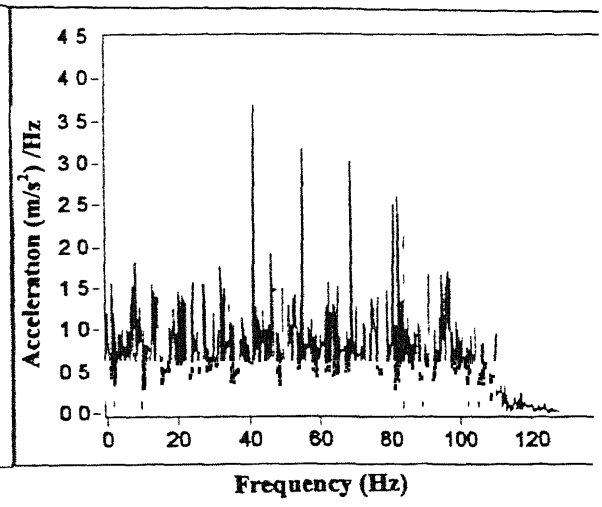


Fig 4 4 (b) Vibration signal in Frequency dom at 800 RPM for Mass unbalance

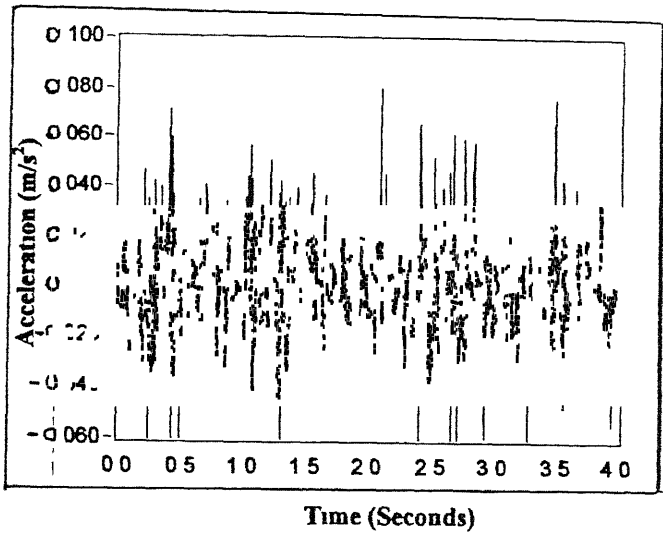


Fig 4 5 (a) Vibration signal in time domain at 800 RPM for Bearing cap loose

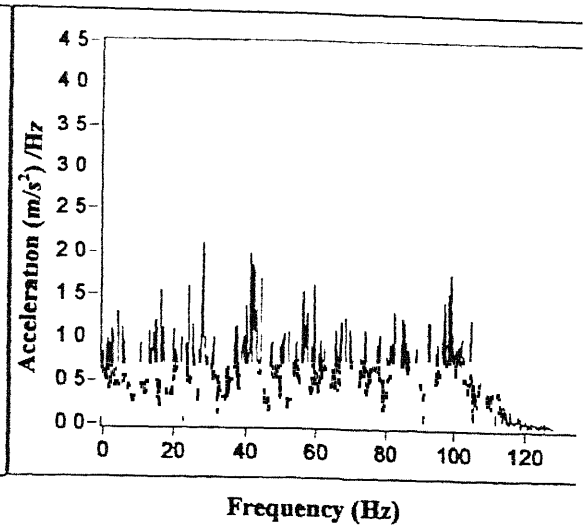


Fig 4.5 (b) Vibration signal in Frequency dom at 800 RPM for Bearing cap loose

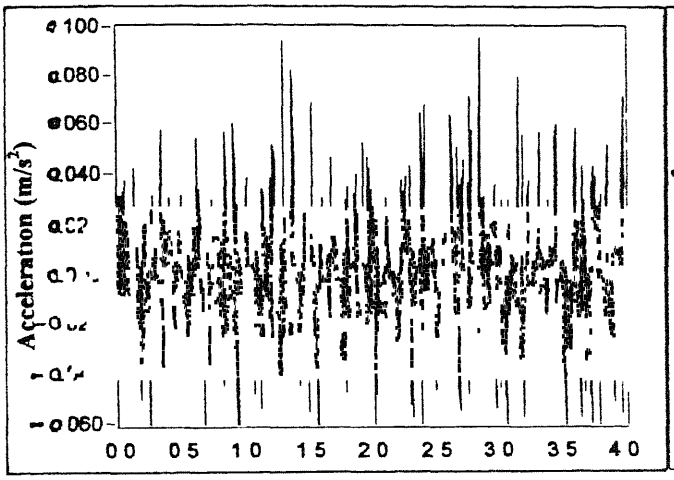


Fig 4 6 (a) Vibration signal in time domain at 800 RPM for Misalignment

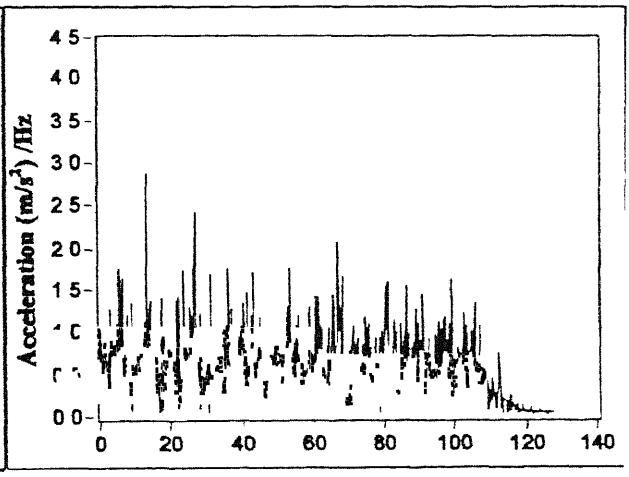


Fig 4 6 (b) Vibration signal in Frequency domain at 800 RPM for Misalignment

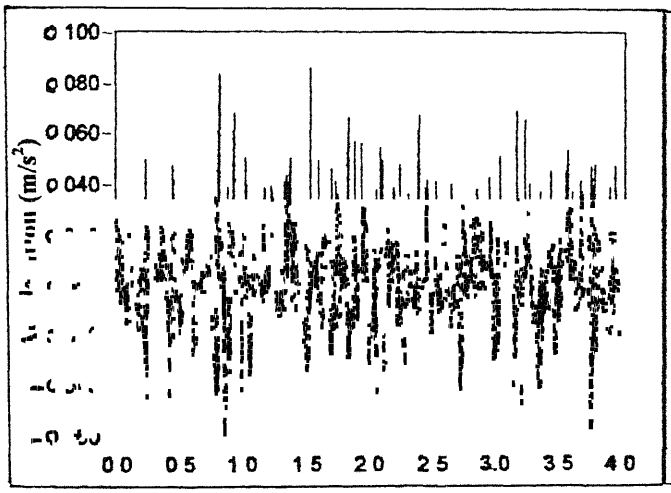


Fig 4.7 (a) Vibration signal in time domain at 800 RPM for Mass unbalance and Misalignment

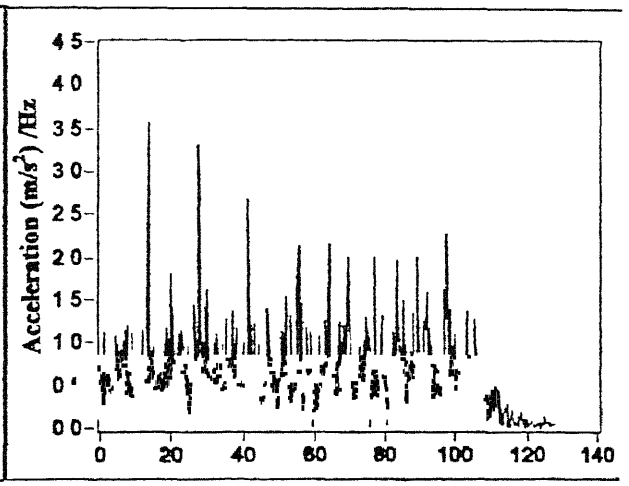


Fig 4.7 (b) Vibration signal in Frequency domain at 800 RPM for Mass unbalance and Misalignment

The mean of the vibration signal is the First Order Moment and is expressed as

$$E\{z(t)\} = \eta = \int_{-\infty}^{\infty} z(t)p(x)d_x \qquad (4\ 3)$$

Two types of Second and Higher Order Moments are taken as follows

$$\text{Central moments } \mu_k = E\{(z(t)-\eta)^k\} = \int_{-\infty}^{\infty} (z(t)-\eta)^k p(x)d_x \qquad k = 1,2,3 \qquad (4\ 4)$$

$$\text{Cross moments (Cross Correlation) } r_{x_1} = \sum_{l=-\infty}^{\infty} a_x(t)a_1(t-l) \qquad l = 0,\pm1,\pm2 \qquad (4\ 5)$$

where, E() is the expectation operator and l is the time-lag

The mean second to fifth central moments of the series $|z(t)|, |dz(t)|$ are calculated at various speeds for all faults introduced The moments thus computed for all the faults introduced are shown as functions of operating speed in Figs 4 8 to 4 10

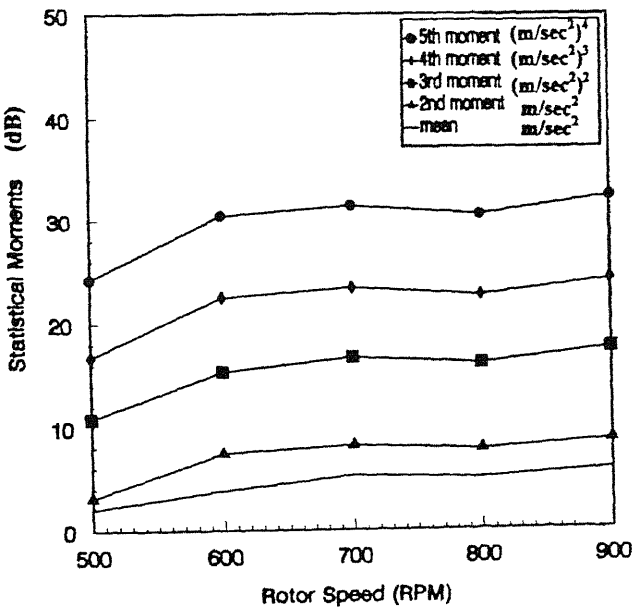


Fig 4 8 (a) **Statistical Moments Of $|z(t)|$ No-fault**

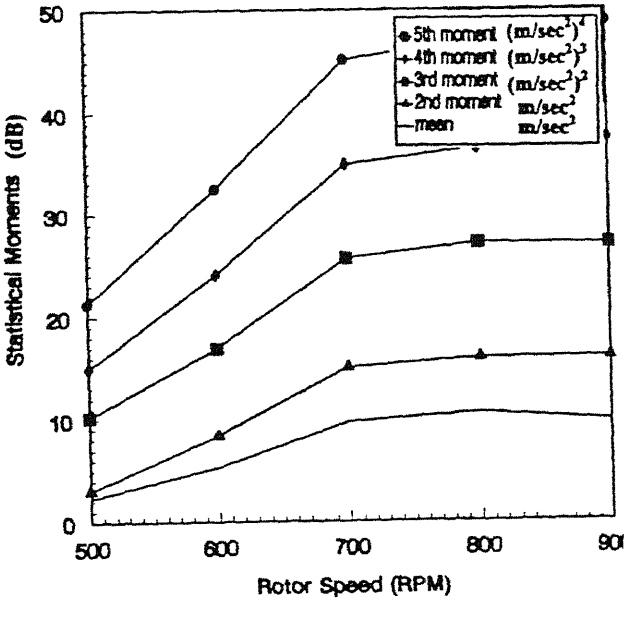


Fig 4 8 (b) **Statistical Moments Of $|z(t)|$ Coupling loose**

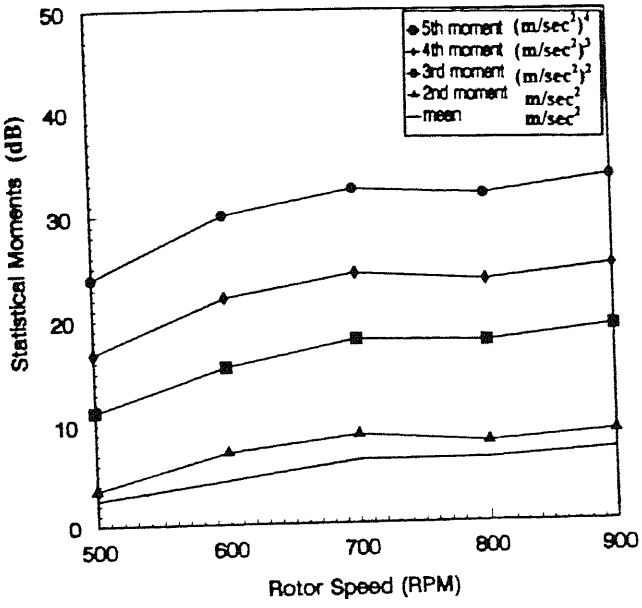


Fig 4 8 (c) Statistical Moments Of $\dot{x}(t)$
Mass unbalance

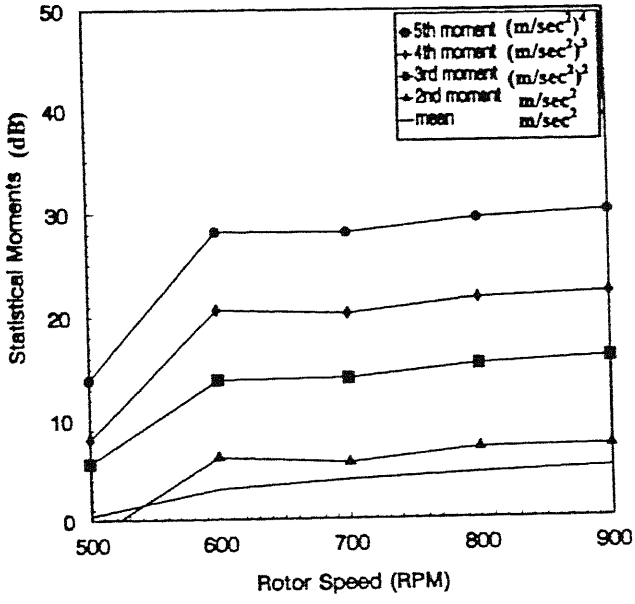


Fig 4 8 (d) Statistical Moments Of $\dot{x}(t)$
Bearing cap loose

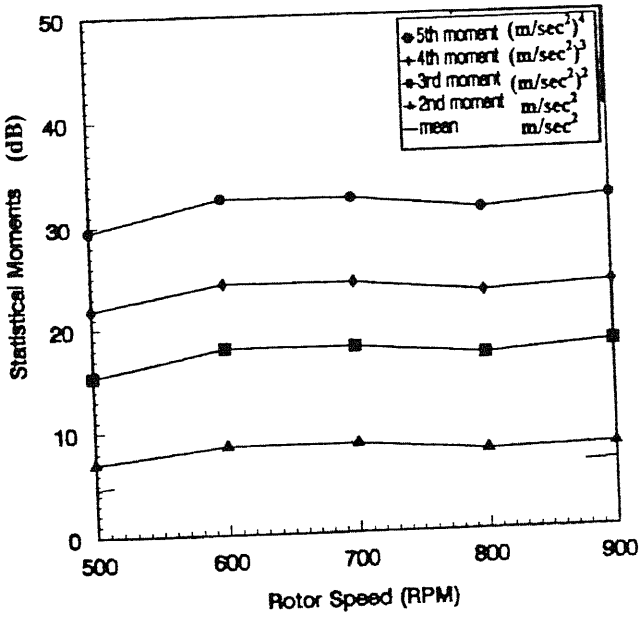


Fig 4 8 (e) Statistical Moments Of $\dot{x}(t)$
Misalignment

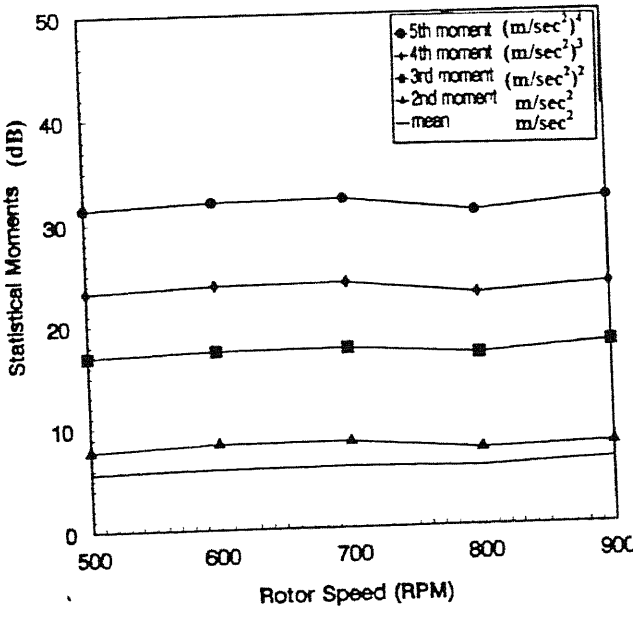


Fig 4.8 (f) Statistical Moments Of $\dot{x}(t)$
Mass unbalance And Misalignment

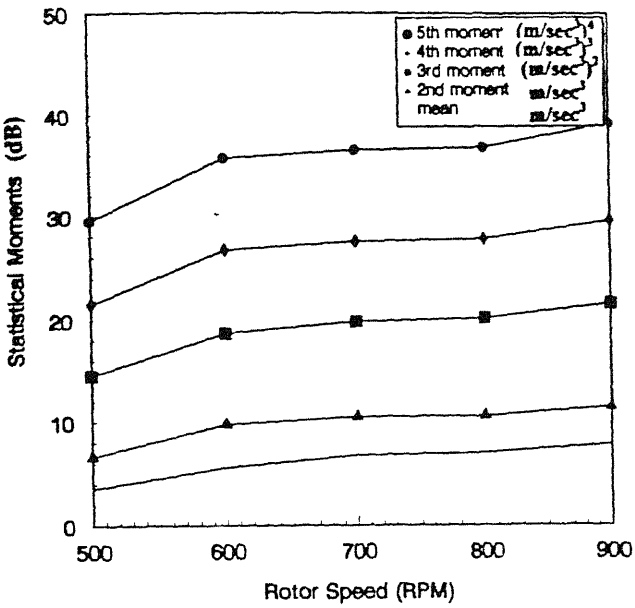


Fig 4.9 (a) Statistical Moments Of $|d\bar{z}(t)|$ No-fault

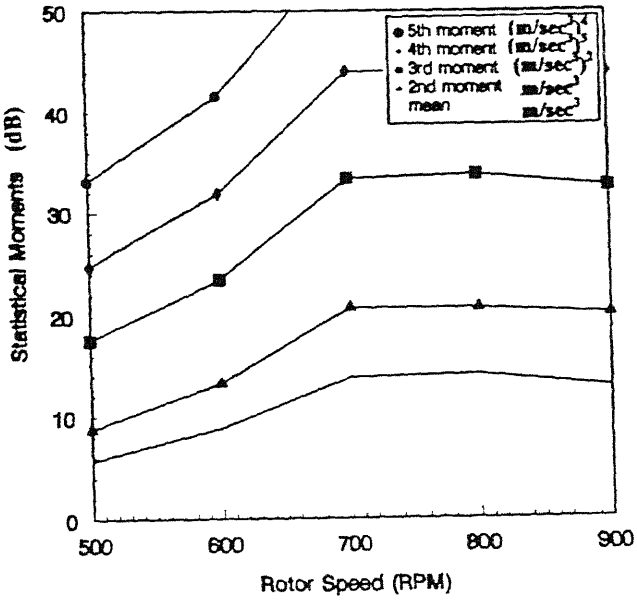


Fig 4 9 (b) Statistical Moments Of $|d\bar{z}(t)|$ Coupling loose

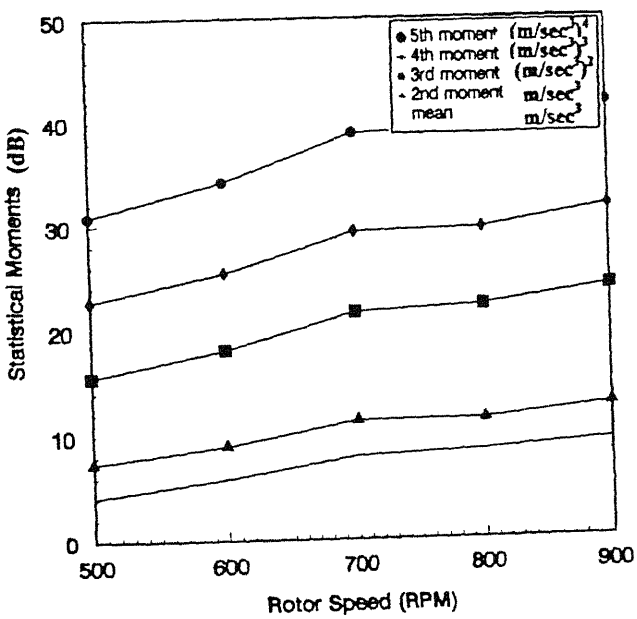


Fig 4.9.(c) Statistical Moments Of $|d\bar{z}(t)|$ Mass unbalance

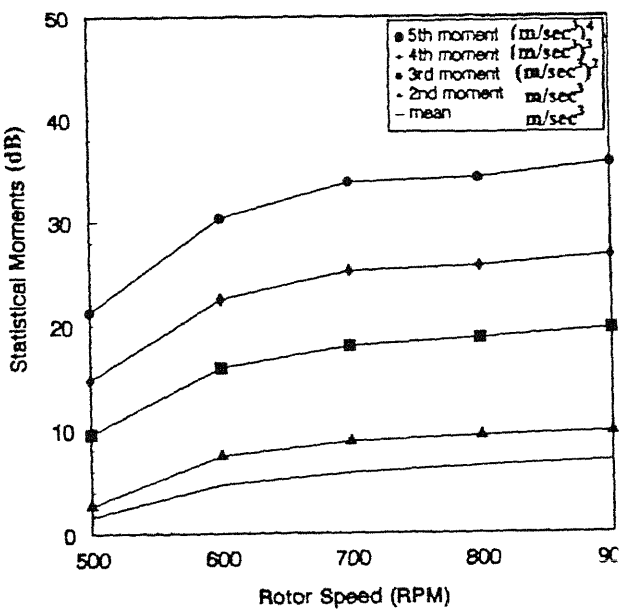


Fig 4 9 (d) Statistical Moments Of $|d\bar{z}(t)|$ Bearing cap loose

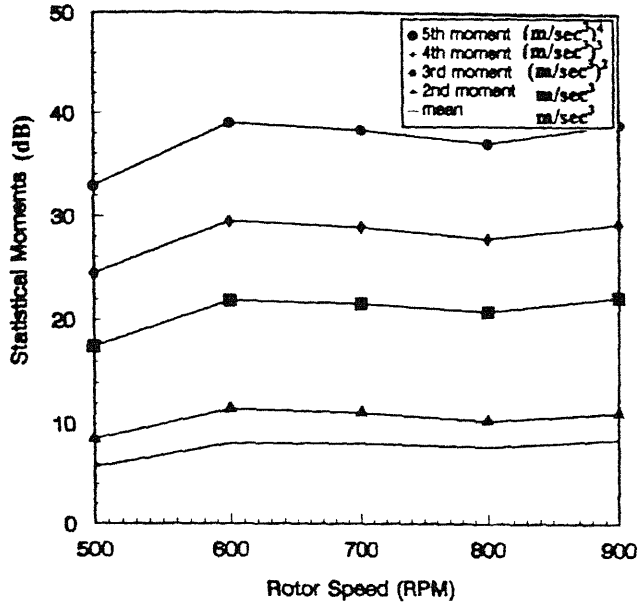


Fig 4 9 (e) Statistical Moments Of $|d\bar{z}(t)|$ Misalignment

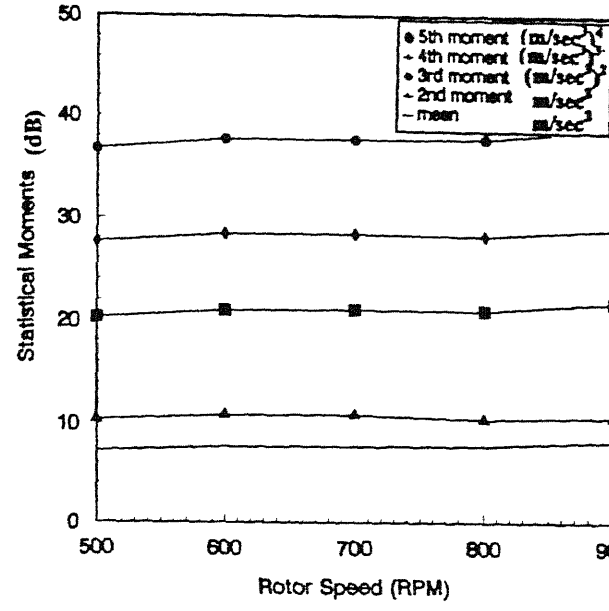


Fig 4 9.(f) Statistical Moments Of $|d\bar{z}(t)|$ Mass unbalance And Misalignment

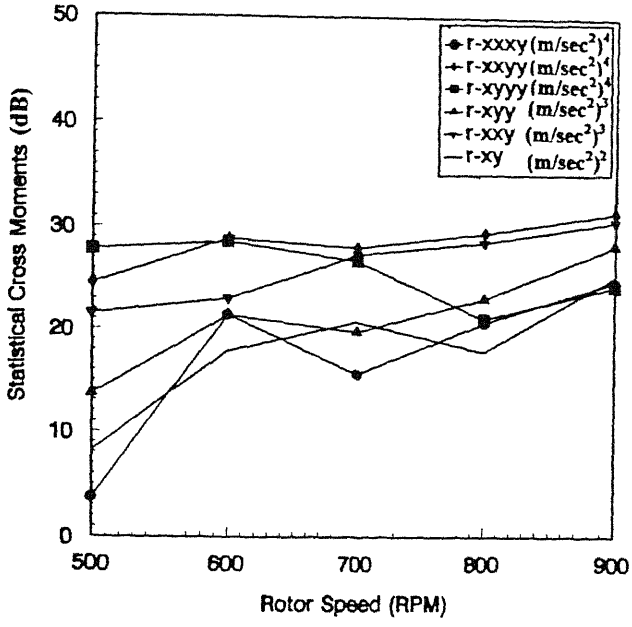


Fig 4.10.(a) Statistical cross Moments No-fault

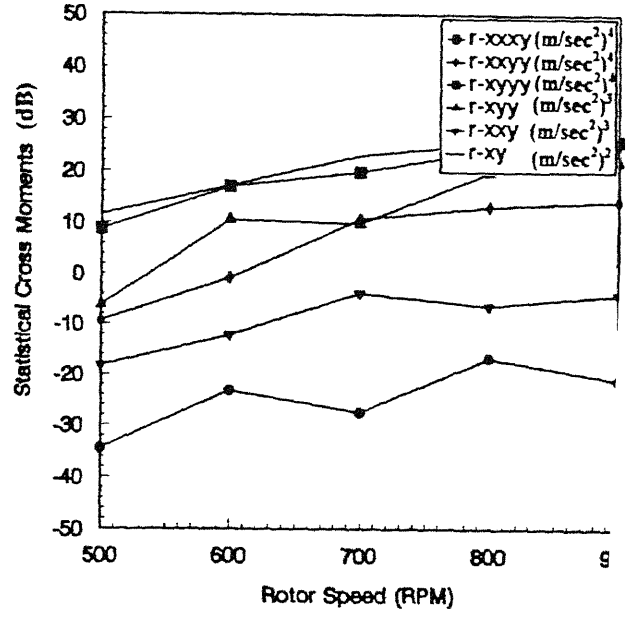


Fig 4.10.(b) Statistical cross Moments Coupling loose

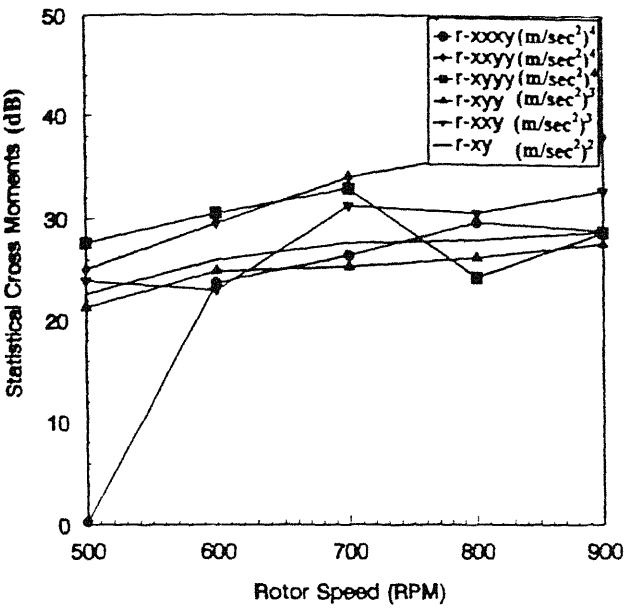


Fig 4.10 (c) Statistical cross Moments
Mass unbalance

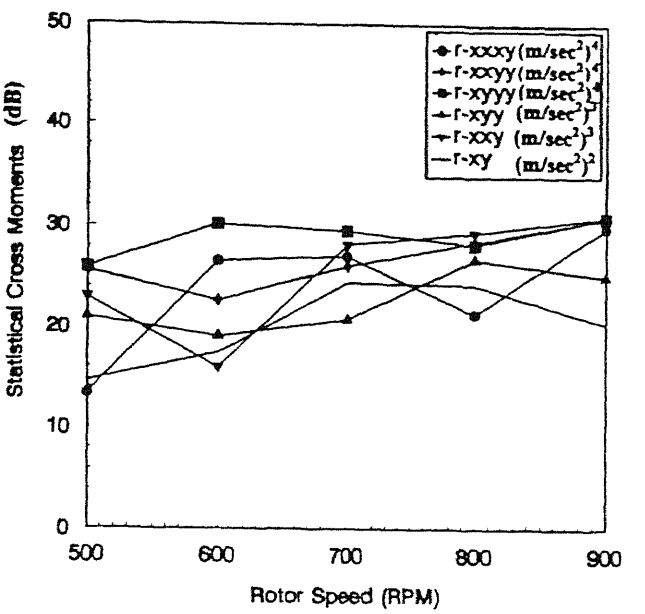


Fig 4.10 (d) Statistical Moments
Bearing cap loose

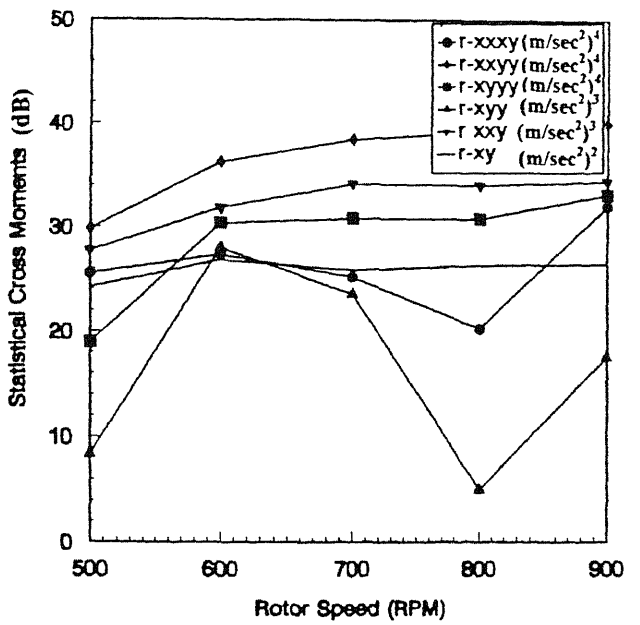


Fig 4.10.(e) Statistical cross Moments
Misalignment

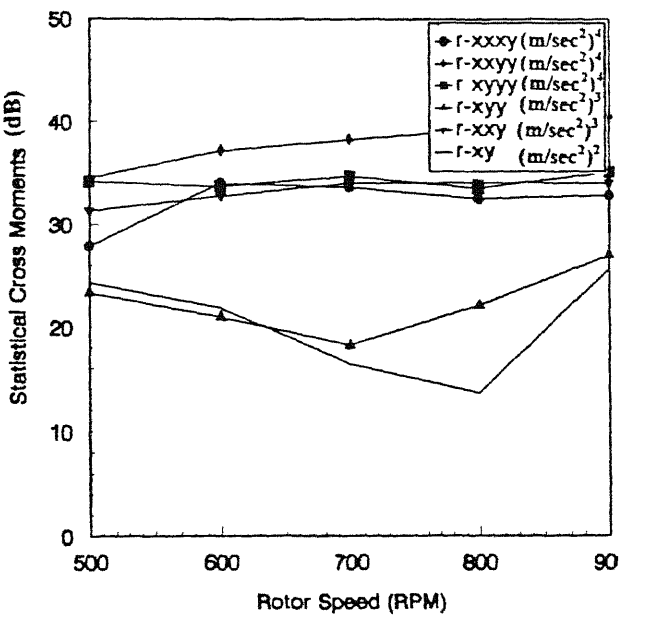


Fig 4.10.(f) Statistical cross Moments
Mass unbalance And Misalignment

4.3 Network Structure

The Neural Network Algorithm is developed for a feed-forward type architecture with one or more hidden layers The architecture of such a network is referred in terms of numerals as $(i,j1,j2, \quad jn,k)$ where i denotes the number of neurons in the input layer jn is the number of neurons in the n-th hidden layer and k denotes the number of output neurons

The moments of the time series defined above form the training attributes to the network They are normalized between 0 1 and 0 9 before feeding them to the network The target output, for training, is made to be a five-dimensional vector The dimension of the target output vector is equal to the number of primary faults for which the data is available Since, five types of primary faults were introduced in the laboratory ing the dimension of the target output vector was made to be five The sixth fault is a combination of mass unbalance and misalignment The target output vector is chosen to have a value equal to either 1 or 0, in each dimension A value equal to 1 in a dimension is symbolic of the presence of a particular fault, while a value equal to zero, signifies its absence

Training is carried out with the following sets of data -

1 Central Moments of $|z(t)|$
2 Central Moments of $|dz(t)|$
3 Combined central moments of $|z(t)|$ and $|dz(t)|$
4 Cross moments of the time series $z(t)$

The Table 4 1 describes the training scheme with the central moments of $|z(t)|$

Table 4.1 Training data

No	Input Vector					Output Vector				
	m1	m2	m3	m4	m5	NF	COU	MU	BCL	MA
1	3 825	5 497	50 216	201 730	1356 723	0 000	0 000	1 000	0 000	1 000
2	9 104	35 538	407 055	3702 857	40905 828	0 000	1 000	0 000	0 000	0 000
3	1 058	0 628	3 292	5 597	19 738	0 000	0 000	0 000	1 000	0 000
4	3 073	5 475	38 027	163 432	974 312	0 000	0 000	0 000	0 000	1 000
5	1 348	1 505	7 460	22 315	91 944	0 000	1 000	0 000	0 000	0 000
6	3 144	5 434	37 044	150 688	858 668	1 000	0 000	0 000	0 000	0 000
7	1 062	0 546	3 242	4 753	17 142	0 000	0 000	0 000	1 000	0 000
8	2 725	5 196	33 661	150 329	866 723	0 000	0 000	1 000	0 000	0 000
9	1 127	0 695	3 972	7 054	26 745	0 000	0 000	0 000	1 000	0 000
10	4 481	7 322	71 231	329 811	2471 105	0 000	0 000	0 000	0 000	1 000
11	4 063	7 876	64 686	338 588	2596 214	0 000	0 000	1 000	0 000	1 000
12	3 338	5 774	40 471	182 959	1142 190	0 000	1 000	0 000	0 000	0 000
13	3 452	5 385	43 947	173 109	1057 272	0 000	0 000	1 000	0 000	1 000
14	4 282	7 427	62 638	267 333	1737 198	0 000	0 000	1 000	0 000	0 000
15	4 920	6 883	70 051	251 523	1715 546	0 000	0 000	1 000	0 000	0 000
16	4 174	7 779	68 666	334 536	2387 193	0 000	0 000	0 000	0 000	1 000
17	2 024	4 243	24 412	120 046	716 274	0 000	0 000	0 000	1 000	0 000
18	4 490	6 139	65 094	253 583	1866 117	0 000	0 000	0 000	0 000	1 000
19	3 669	5 560	48 572	204 294	1369 507	0 000	0 000	1 000	0 000	1 000
20	3 605	5 023	43 976	166 848	1082 105	0 000	0 000	0 000	0 000	1 000

m1-m5 moments

NF No-fault **COU** Coupling loose **MU** Mass unbalance

BCL Bearing cap loose **MA** Misalignment

Six hundred such data are used for training the network Initially all the weights in the network are adjusted between -0 5 and 0 5 randomly Once the learning process starts, the neural network is so designed that the weights and the thresholds between different layers adjust automatically, so as to minimize the mean square error, between the actual network output and the targeted output In minimizing the error, the other network parameters like moment, learning rate, number of hidden layers and nodes in each layer are adjusted The output vectors which are generated by the network do not always consists of exactly either 0 or 1 If a dimension value is greater than or equal to 0 6 it is considered as 1 Similarly, if it is less than or equal to 0 3 it is considered as 0 If the vector consists of a quantity less than 0 6 in all dimension the prediction of fault is ambiguous The network algorithm is trained for different architectures and training schemes

4.4. Network Training With First Five Moments Of $|z(t)|$

The training process was started with the mean and the second to fifth central moments of $|z(t)|$ The network needs to (5,$j1$,$j2$, jn,5) architecture - since there are five input nodes, each node corresponding to a moment input, and an output layer consisting of 5 nodes, each node corresponding to one of the five primary faults under consideration The number of hidden layers can be varied The number of neurons in each hidden layer can also be varied A total of 600 samples, comprising of 100 samples for each fault, was used for the training the network A numerical value is required initially, to quantitatively define the accuracy of the training process A training process will be successful if it is able to correctly identify the fault The degree of such accuracy, called the target error, is the mean square difference between a target value (0 or 1) and the achieved output To start with, the achievable accuracy was arbitrarily chosen as 0 01 A network with two hidden layers, each with five neurons (network architecture 5,5,5,5) was tried The convergence to the value of 0 01 was not achieved even after 2000 iterations (see the error graph of Fig 4 11) The number of hidden layers and the number of neurons in each layer was varied The mean square error achievable was in the vicinity of 0 7 When the network, thus trained (to a mean square error accuracy of 0 7), was given a fresh set of inputs, obtained from various faulty vibration signatures (faults being unknown to the network, since these input sets did not form a part of the training data), it was found that the network had failed to correctly identify the faults in a significant way The testing success was only between 35 to 45% The performance of the various network architectures (for learning rate 0 2 and moment 0 8) is shown in Table 4 2 (about 15 such architectures were tried, five best are given in the table) The general observation was that the overall performance of the network with first five moments of the $|z(t)|$ was between 35 to 45%

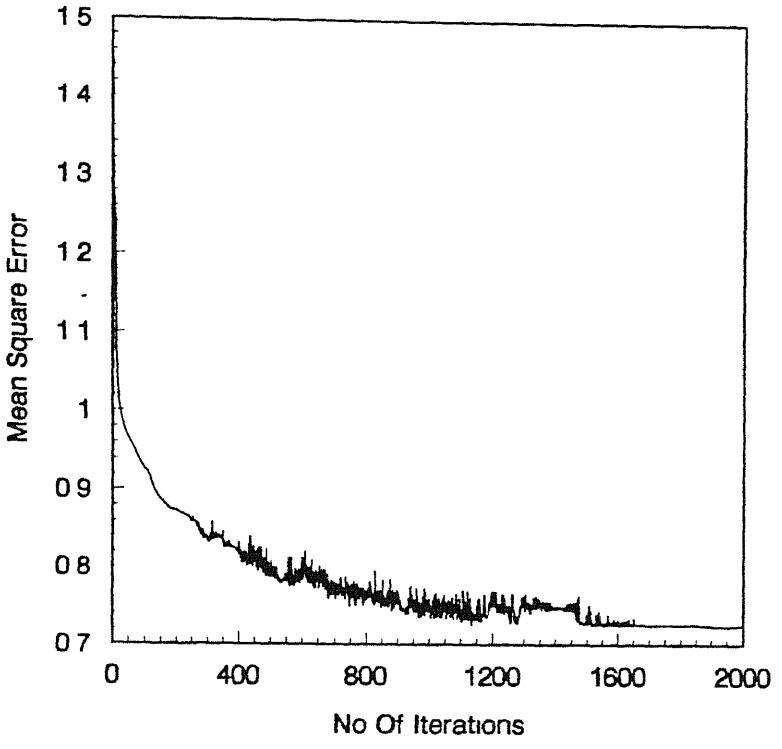


Fig 4 11 Error graph. for the network with 5 moments

Table 4.2. Different network architectures and testing success for the network with first five moments of $|z(t)|$

No	Network Architecture	Target Error	Achieved Error	Iterations	Testing Success
1	5,5,5,5	0 01	0 7374	2000	35 0%
2	5,8,8,5	0 01	0 6118	2000	42 5%
3	5,6,6,5	0 01	0 6713	2000	41 0%
4	5,10,5	0 01	0 6996	2000	47 5%
5	5,6,6,5	0 01	0 7043	2000	42 5%

4.5. Network Training With First Five Moments Of $|dz(t)|$

The training was carried in a manner similar to that adopted for the moments of $z(t)$ The mean, second to fifth central moment of $|dz(t)|$ were used as inputs The training was initially carried out with a set of 600 samples and a target error of 0 01 Convergence

could not be achieved in this case also The error was never below 0 6 for all architectures that were trained and tested The success rate was 50 to 60% The different network architectures and testing success are shown in Table 4 3 (about 15 such architectures were tried, five best are given in the table)

Table 4.3. Different network architectures and testing success for the network with first five moments of $|dz(t)|$

No	Network Architecture	Target Error	Achieved Error	Iterations	Testing Success
1	5,5,5	0 01	0 7613	2000	50 0%
2	5,10,5	0 01	0 7109	2000	52 5%
3	5,10,5	0 01	0 7004	2000	57 5%
4	5,8,8,5	0 01	0 7426	2000	57 5%
5	5,6,6,5	0 01	0 7446	2000	60 0%

4.6. Network Training With Moments Of both $|z(t)|$ $|dz(t)|$

An attempt was made to improve the convergence of network by performing the training with the moments of $|z(t)|$ and $|z(t)|$ simultaneously The training was carried out for a (10,10,5) architecture (the input layer now has 10 neurons since there are 10 input vectors) with a target error of 0 01 It was observed that the mean square decreased to about 0 46 which was lower than the error obtained with moments of $|z(t)|$ alone or $|z(t)|$ alone On testing the network thus trained it was found that it was able to correctly identify the faults in about 60% of the cases Table 4 4 gives the results obtained for such tests

Table 4.4. Performance of (10,10,5) architecture

No	Network Architecture	Moment	Learning Rate	Target Error	Achieved Error	Iterations	Testing Success
1	10,10,5	0 90	0 10	0 01	0 4611	2000	60 00%
2	10,10,5	0 70	0 20	0 01	0 5022	2000	61 25%
3	10,10,5	0 80	0 10	0 01	0 4055	2000	65 00%
4	10,10,5	0 90	0 20	0 01	0 4628	2000	65 00%
5	10,10,5	0 80	0 25	0 01	0 4823	2000	68 67%

Since the results were encouraging an additional hidden layer with 10 neurons was introduced The architecture now was (10,10 10,5) The performance of this architecture was noted for various learning rates Fig 4 12 graphically shows the decreasing error trends for (10,10,10,5) architecture for two different learning rates A learning rate of 0 25 generated the highest success of 83% The performance of this training scheme is given in Table 4 5

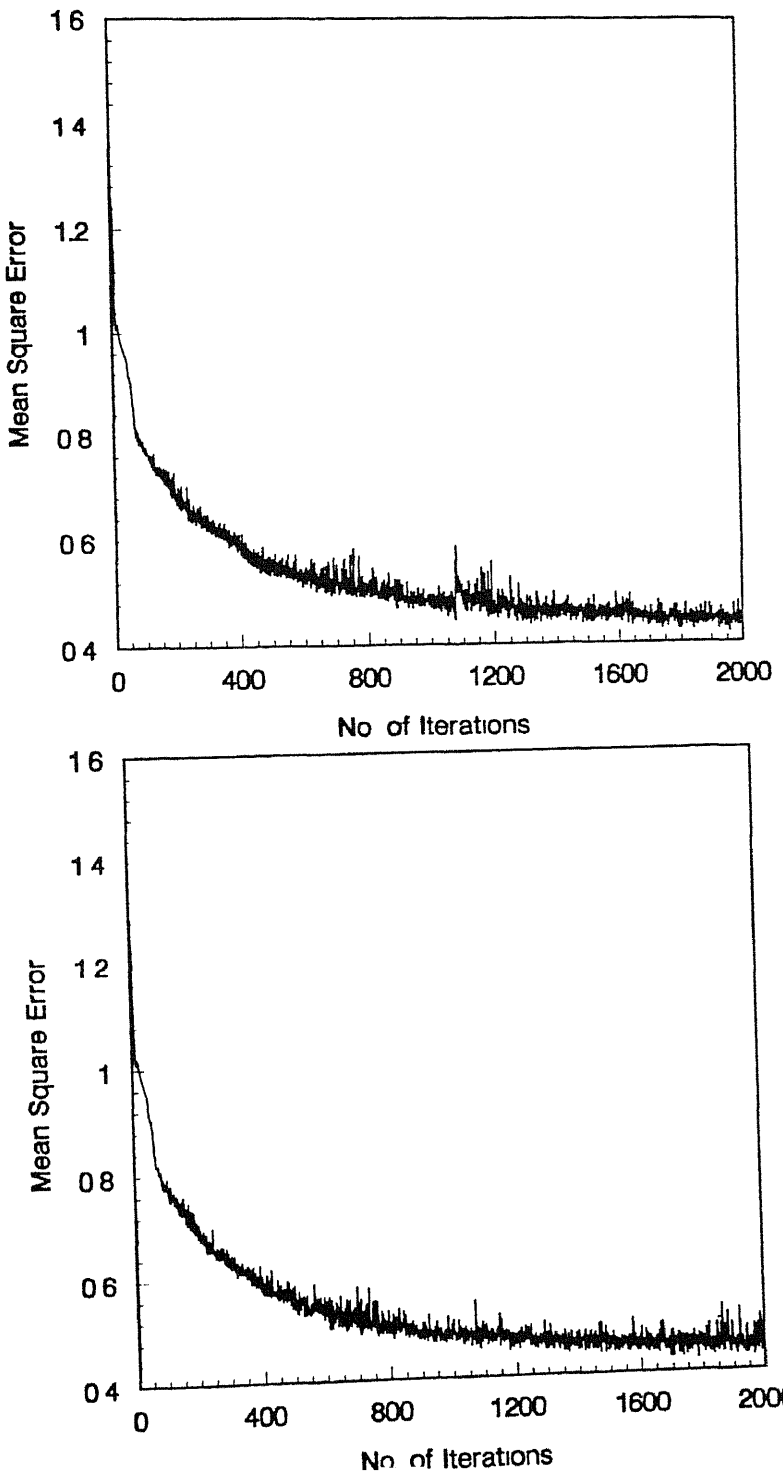


Table 4.5. Comparison of network output with actual output for the network with 10 inputs

No	Network output					Actual Output				
	NF	COU	MU	BCL	MA	NF	COU	MU	BCL	MA
1	0 009	0 001	0 065	0 677	0 064	0 000	0 000	0 000	1 000	0 000
2	0 000	0 003	0 595	0 023	0 860	0 000	0 000	0 000	0 000	1 000
3	0 003	0 000	0 058	0 000	0 832	0 000	0 000	0 000	0 000	1 000
4	0 000	1 000	0 001	0 000	0 000	0 000	1 000	0 000	0 000	0 000
5	0 232	0 000	0 222	0 077	0 000	0 000	0 000	0 000	1 000	0 000
6	0 018	0 001	0 038	0 827	0 029	0 000	0 000	0 000	1 000	0 000
7	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
8	0 006	0 003	0 640	0 004	0 710	0 000	0 000	0 000	0 000	1 000
9	0 953	0 003	0 048	0 001	0 016	1 000	0 000	0 000	0 000	0 000
10	0 000	1 000	0 001	0 000	0 000	0 000	1 000	0 000	0 000	0 000
11	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
12	0 995	0 002	0 014	0 003	0 002	1 000	0 000	0 000	0 000	0 000
13	0 030	0 002	0 663	0 001	0 598	0 000	0 000	1 000	0 000	1 000
14	0 000	0 000	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000
15	0 093	0 002	0 679	0 001	0 510	0 000	0 000	1 000	0 000	1 000
16	0 090	0 002	0 679	0 001	0 513	0 000	0 000	1 000	0 000	1 000
17	0 000	1 000	0 001	0 000	0 000	0 000	1 000	0 000	0 000	0 000
18	0 081	0 002	0 677	0 001	0 521	0 000	0 000	1 000	0 000	0 000
19	0 009	0 000	0 944	0 005	0 000	0 000	0 000	1 000	0 000	0 000
20	0 034	0 871	0 021	0 001	0 041	0 000	1 000	0 000	0 000	0 000
21	0 006	0 901	0 080	0 107	0 200	0 000	1 000	0 000	0 000	0 000
22	0 193	0 000	0 025	0 592	0 000	0 000	0 000	0 000	1 000	0 000
23	0 012	0 760	0 032	0 213	0 045	0 000	1 000	0 000	0 000	0 000
24	0 300	0 103	0 400	0 600	0 000	0 000	0 000	0 000	1 000	0 000
25	0 502	0 042	0 312	0 087	0 803	0 000	0 000	0 000	0 000	1 000

4.7. Network Training With Cross Moments

Training was also attempted for sets of data comprising of the mean, the second, third and fourth order central moments of horizontal and vertical displacements along with six cross moments upto the fourth order The number of input nodes in this case becomes 14, while the output nodes remain equal to 5 Architectures with two hidden layers were trained and tested The number of neurons in each hidden layer was kept variable It was found that the best results were obtained with 14 neurons in each hidden layer Increasing the number of neurons further resulted in overtraining and decreased the success rate The learning rate and moment were now varied and the network performance was recorded Table 4 6 gives this record

Table 4.6. Different network architectures and testing success for the network with 14 input

No	Network Architecture	Moment	Learning Rate	Target Error	Achieved Error	Iterations	Testing Success
1	14,14,14 5	0 50	0 15	0 01	0 0334	2000	80 00%
2	14,14,14,5	0 90	0 15	0 01	0 0697	2000	80 00%
3	14,14,14,5	0 90	0 10	0 01	0 0670	2000	82 85%
4	14,14,14,5	0 98	0 20	0 01	0 1567	2000	85 71%
5	14,14,14,5	0 90	0 20	0 01	0 1252	2000	87 14%
6	14,14,14,5	0 90	0 40	0 01	0 1813	2000	88 57%
7	14,14,14,5	0 85	0 15	0 01	0 1005	2000	90 00%
8	14,14,14,5	0 80	0 10	0 01	0 0545	2000	91 42%

It can be seen from Table 5 5 that for a value of moment equal to 0 8 and learning rate equal to 0 1, the network converges to a mean square error of 0 05 The error graph (Fig 4 13) shows the decreasing error with the number of iterations performed The success rate in this case is found to be 91% A comparison of the network output and the actual output for fault identification in this case is shown in Table 4 7

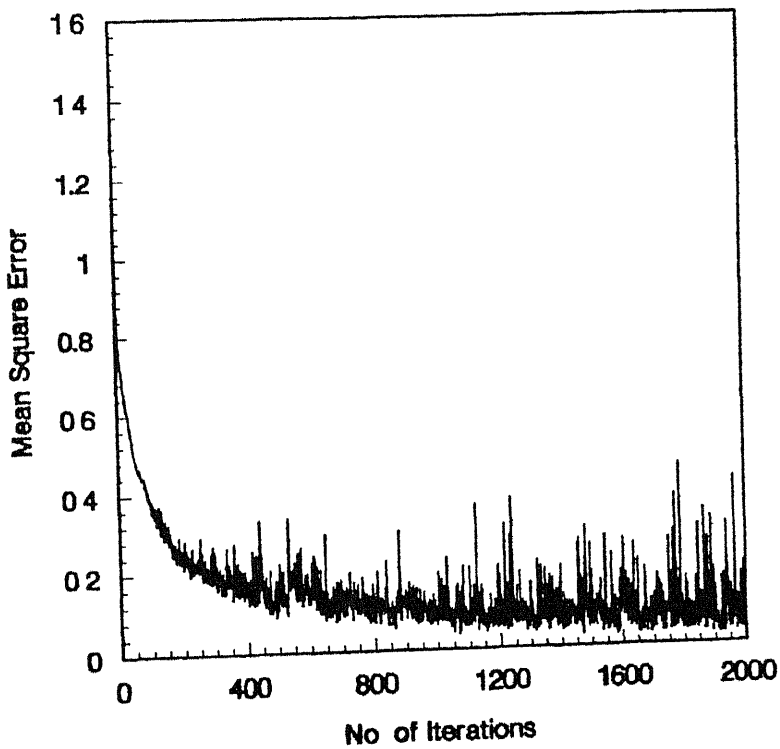


Fig 4 13 Error graph for the network with 14 moments

Table 4.7. Comparison of network output with actual output for the network with 14 inputs

No	Network Output					Actual Output				
	NF	COU	MU	BCL	MA	NF	COU	MU	BCL	MA
1	0 000	0 000	0 000	0 999	0 000	0 000	0 000	0 000	1 000	0 000
2	0 000	0 000	0 001	0 013	1 000	0 000	0 000	0 000	0 000	1 000
3	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000	0 000	1 000
4	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
5	0 000	0 000	0 000	0 999	0 000	0 000	0 000	0 000	1 000	0 000
6	0 000	0 079	0 000	0 960	0 000	0 000	0 000	0 000	1 000	0 000
7	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
8	0 000	0 000	0 157	0 002	1 000	0 000	0 000	0 000	0 000	1 000
9	0 602	0 000	0 409	0 000	0 000	1 000	0 000	0 000	0 000	0 000
10	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
11	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
12	0 001	0 003	0 000	1 000	0 000	1 000	0 000	0 000	0 000	0 000
13	0 000	0 000	0 984	0 000	1 000	0 000	0 000	1 000	0 000	1 000
14	0 000	0 003	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000
15	0 000	0 000	0 766	0 000	0 996	0 000	0 000	1 000	0 000	1 000
16	0 000	0 000	0 700	0 006	1 000	0 000	0 000	1 000	0 000	1 000
17	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
18	0 004	0 000	0 005	0 938	0 000	0 000	0 000	0 000	1 000	0 000
19	0 025	0 000	0 610	0 431	0 341	0 000	0 000	1 000	0 000	0 000
20	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
21	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
22	0 010	0 000	0 000	0 987	0 054	0 000	0 000	0 000	1 000	0 000
23	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000
24	0 000	0 003	0 000	1 000	0 000	0 000	0 000	0 000	1 000	0 000
25	0 000	0 000	0 000	0 000	1 000	0 000	0 000	0 000	0 000	1 000

4.8. Remarks

It was observed from the above training schemes that, the network had failed to identify the *mass unbalance* In some cases, *mass unbalance* was identified as a combination of *mass unbalance* and *misalignment* In all cases the network successfully identified the samples that are fault-free It was also observed that *bearing cap loose* case and *play in coupling* were identified very accurately by the network In some cases network had minor problems in identifying the combination of *mass unbalance* and *misalignment* The overall performance of the network can be said to lie approximately between 75 to 85%

CONCLUSIONS

The present study needs to be viewed as a preliminary attempt at making use of the Artificial Neural Network technology for fault prediction in rotating machinery During the course of this work focus was kept on the most crucial aspect of data generation for healthy and faulty rotor system The well established technique of using statistical moments of the system signatures for network training were adopted Faults were deliberately introduced in the rotor system and vibration signals were picked up using piezo-electric sensors The vibration signals were then statistically processed to obtain the moments to be fed as inputs to the training algorithm The ANN simulator was written in ANSI-C, trained, tested and validated The results obtained are encouraging and lead to optimism on the possible use of ANNs as an effective tool for rotating machinery fault diagnosis This would, however, require a more rigourous training environment - which could vary the training parameters like learning rate, moment automatically or in an interactive manner and also make a comparison between the effectiveness of various sets of inputs, to obtain best possible results All this can be taken up as future development of the present work The basic framework of the algorithm can remain the same Additiona modifications and alterations can be made wherever necessary The simulator may need to be refined by additional experimental data generation also The possibility of incorporating the present algorithm, which is presently written in ANSI-C, as a Virtual Instrument icon in a Virtual Instrumentation Software is very attractive This can result in a package which combines data acquisition, processing and fault prediction through an ANN

REFERENCES

1 Ahn K G , Pahk H J , Jung M Y and Cho D W A hybrid-type active vibration isolation system using neural networks ***Journal of Sound and Vibration*** Vol 192, No 4, 1996

2 Childs D ***Turbomachinery Rotordynamics*** John Wiley & Sons 1993

3 Dhingra A K and Rao S S A neural network based approach to mechanical design optimization ***Engineering Optimization*** Vol 20, 1992

4 Dimarogonas A D and Paipetis S A ***Analytical Methods In Rotor Dynamics*** Applied science publishers 1983

5 Dimarogonas A D Fuzzy input-neural net adoptive expert systems for rotor diagnosis and prognasis 1995

6 Dimentberg M F ***Statistical Dynamics Of Nonlinear And Time-Varying Systems*** John Wiley 1988

7 Elkordy M F , Chang K C and Lee G C Application of Neural Networks in Vibration Signature Analysis ***Journal of Engineering Mechanics*** Vol 120, No 2 February 1994

8 Ehrich F F ***Handbook Of Rotordynamics*** McGraw-Hill Inc 1992

9 Kramer E ***Dynamics Of Rotors And Foundations*** Springer-Verlag 1993

10 Han Y , Xiu L , Wang Z , Chen Q , Tan S Artificial neural network controlled fas valving in a power generation plant ***IEEE Transactions on neural networks*** Vol 8 No 2, March 1997

11 Haykin S ***Neural Networks. A comprehensive foundation*** Macmillan New Yorl 1994

12 Hebb D O Organization of behavior ***Science Editions*** New York 1961

13 Huang S J and Lian R J Acombination of fuzzy logic and neural networ algorithms for active vibration control ***Proceedings Institute Mechanical Engineers*** Vol 210, 1996

14 Kram M , Ghassemzadeh M , Dai N , Gandikota M , Trzyanadlowski A M Validation and recovery of vibration data in electromachine systems using neur network software ***IEEE Transactions on industrial applications*** Vol 30, No November/December 1994

15 Mayes I W Use of neural network for on-line vibration monitoring ***Proceedings Institute of mechanical engineers*** Vol 208 1994

16 McCormick A C and Nandi A K Real time classification of Rotating shaft loadi conditions using Artificial neural networks ***IEEE Transactions on neural networ*** Vol 8, No 3, May 1997

17 Nema N ***Expert systems for fault diagnosis.*** M-tech thesis report (ME) 1994

18 Newland D E ***An introduction to Random vibrations and Spectral analys*** Longman Scientific and technical New York 1984

19 Papoulis A ***Probability, Random variables and stochastic processes*** McGraw-Hi Inc 1991

20 Proakis J G and Manolakis D G ***Digital Signal Processing Princip Algorithms And Applications*** Prentice-Hall, India 1997

21 Rao J S ***Rotor Dynamics*** Wiley Eastern Ltd, 1992

22 Wassermann P D ***Neural Computing Theory And Practice*** Van Nostrand einhc New York 1990

23 Yen G G and Kwak M K Neural Network approach for the damage detection structures ***American Institute of Aeronautics and Astronautics*** 1993